

**ANÁLISIS, MANTENIMIENTO, DISEÑO, DESARROLLO E IMPLEMENTACIÓN  
DE NUEVAS FUNCIONALIDADES Y MEJORAS RELACIONADAS CON LOS  
MÓDULOS DE PORTAL WEB DE GRUPOS Y TRABAJOS DE GRADO EN LA  
PLATAFORMA COMA COMUNIDAD ACADÉMICA.**

**ALEXY YESSENIA RINCÓN CAPERA**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS  
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA  
BUCARAMANGA**

**2019**

**ANÁLISIS, MANTENIMIENTO, DISEÑO, DESARROLLO E IMPLEMENTACIÓN  
DE NUEVAS FUNCIONALIDADES Y MEJORAS RELACIONADAS CON LOS  
MÓDULOS DE PORTAL WEB DE GRUPOS Y TRABAJOS DE GRADO EN LA  
PLATAFORMA COMA COMUNIDAD ACADÉMICA.**

**ALEXY YESSENIA RINCÓN CAPERA**

**Trabajo de grado para optar al título de  
Ingeniera de Sistemas**

**Director**

**Msc. LUIS IGNACIO GONZÁLEZ RAMÍREZ  
Magíster en Informática**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FISICOMECAÑICAS  
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA  
BUCARAMANGA  
2019**

## **AGRADECIMIENTOS**

A mis padres María Doris Capera Díaz y Gustavo Rincón Solano, a mi Hermano Daniel Fernando Rincón Capera, comprensión y apoyo durante mi proceso de formación profesional y personal.

Al profesor Luis Ignacio por su confianza y apoyo durante la realización de este proyecto de grado.

A Karen Ximena Orozco Gamboa por su apoyo incondicional durante todo este proceso, y a mis compañeros Andrés Felipe Vesga, Carlos Antonio Díaz, Harold Ardila y Tania Estephany Vergara por el apoyo brindado durante el proceso académico.

## CONTENIDO

	<b>pág.</b>
INTRODUCCIÓN	11
1.1    DEFINICIÓN DE LA SITUACIÓN ACTUAL	12
1.2    JUSTIFICACIÓN	13
1.3    OBJETIVOS	15
1.4    IMPACTO Y VIABILIDAD	16
2.    MARCO TEÓRICO	17
2.1    ARQUITECTURA CLIENTE/SERVIDOR	17
2.2    TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS	22
2.3    BASES DE DATOS	25
2.4    NETBEANS	28
2.5    SISTEMA DE CONTROL DE VERSIONES	29
2.6    PROGRAMACIÓN UTILIZADA	30
2.7    SERVIDORES WEB	32
3.    MARCO METODOLÓGICO	34
3.1    PROTOTIPO EVOLUTIVO	34
3.2    LENGUAJE DE MODELADO UNIFICADO	36
3.3    ESTÁNDARES DE PROGRAMACIÓN	40
4.    DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO	42
4.1    PROTOTIPO ESPERADO	42
4.2    DIAGRAMAS DE CASOS DE USO	46
4.8    MANTENIMIENTO Y ADMINISTRACIÓN	60
5    PRUEBAS DEL SISTEMA	61
5.1    PRUEBAS DE VERIFICACIÓN	61
6.    CONCLUSIONES	67
7.    RECOMENDACIONES	69
BIBLIOGRAFÍA	70
ANEXOS	72

## LISTA DE TABLAS

pág.

Tabla 1 Casos de uso: Solicitar retiro al comité de trabajo de grado como director. .....	50
Tabla 2 Casos de uso: Actualizar portal web de grupos. ....	51
Tabla 3. Gestionar visitas.....	52
Tabla 4 Casos de uso: Administrar inscripciones escuelas por un administrador e integrante del grupo. ....	52
Tabla 5 Casos de uso: Inscripciones abiertas escuelas por un integrante del grupo y usuario externo. ....	53
Tabla 6 Casos de uso: Administrar inscripciones comunidad UIS por un administrador e integrante del grupo. ....	53
Tabla 7 Caso de uso: Ver inscritos escuela y ver inscritos UIS por el administrador e integrante del grupo .....	54
Tabla 8 Casos de uso: Inscripciones abiertas comunidad UIS por un integrante del grupo y usuarios de la o las escuelas a las que valla dirigidas dicha inscripción...55	55
Tabla 9. Descripción de las entidades. ....	58
Tabla 10. Pruebas realizadas: Solicitar retiro como director de un proyecto. ....	62
Tabla 11. Pruebas realizadas: Actualizar portal web grupos. ....	63
Tabla 12. Pruebas realizadas: Administrar inscripciones escuelas.....	63
Tabla 13. Pruebas realizadas: Inscripciones abiertas escuelas.....	64
Tabla 14 Pruebas realizadas: Administrar inscripciones comunidad UIS .....	64
Tabla 15. Pruebas realizadas: Ver inscritos comunidad UIS y Escuela en grupos. .....	65
Tabla 16. Pruebas realizadas: Inscripciones abiertas comunidad UIS .....	66

## LISTA DE FIGURAS

	<b>pág.</b>
Figura 1. Modelo Cliente/Servidor.....	18
Figura 2. Modelo de acceso a JSP .....	24
Figura 3. Prototipo evolutivo .....	34
Figura 4. Diagramas de casos de uso. ....	37
Figura 5. Diagramas de secuencias.....	39
Figura 6. Diagrama de casos de uso: Solicitar retiro como director al comité de trabajos de grado. ....	46
Figura 7. Diagrama de casos de uso: Actualizar portal web de grupos. ....	47
Figura 8. Gestionar visitas. ....	47
Figura 9. Diagrama de casos de uso: Administrar inscripciones Escuelas por un administrador e integrante del grupo. ....	48
Figura 10. Diagrama de casos de uso: Inscripciones abiertas Escuelas por un integrante del grupo y un usuario externo.....	48
Figura 11. Diagrama de casos de uso: Administrar inscripciones UIS por un administrador e integrante del grupo. ....	49
Figura 12. Diagrama de casos de uso: Ver inscritos escuela y ver inscritos comunidad UIS por un administrador e integrante del grupo.....	49
Figura 13. Diagrama de casos de uso: Inscripciones abiertas comunidad UIS por un integrante del grupo y un usuario UIS.....	50
Figura 14. Diagrama E/R: Portal de grupos e inscripciones grupos.....	56
Figura 15. Diagrama E/R: Inscripciones comunidad UIS de grupos. ....	57
Figura 16. Diagrama E/R: Solicitar retiro como director de proyecto. ....	57
Figura 17. Diagrama de secuencia: Solicitar retiro como Director. ....	72
Figura 18. Diagrama de secuencia: Retiro como Director sesión del Comité. ....	73
Figura 19. Diagrama de secuencia: Configuración página de inicio. ....	75
Figura 20. Diagrama de secuencia: Gestionar visitas.....	77
Figura 21. Diagrama de secuencia: Inscripciones portal de grupos escuela y UIS	78
Figura 22. Diagrama de secuencia: Inscripciones abiertas por un usuario de la comunidad de la escuela y comunidad UIS .....	78
Figura 23. Diagrama de secuencia: Inscripciones abiertas para integrante del grupo.....	78
Figura 24. Diagrama de secuencia: Ver inscritos por integrante del grupo creador de la inscripción. ....	78
Figura 25. Ver inscritos escuela y ver inscritos UIS por el administrador e integrante del grupo. ....	78

## LISTA DE ANEXOS

	<b>pág.</b>
Anexo A. Modelo de Procesos del Sistema	72

## RESUMEN

**TÍTULO:** Análisis, mantenimiento, diseño, desarrollo e implementación de nuevas funcionalidades y mejoras relacionadas con los módulos de Portal Web de Grupos y trabajos de grado en la plataforma COMA Comunidad Académica. \*

**AUTOR:** Alexy Yessenia Rincón Capera\*\*

**PALABRAS CLAVE:** Portal, Grupos, Trabajos.

## DESCRIPCIÓN

En la actualidad las escuelas cuentan con un sistema de información orientado a la Web que se encarga de la administración y control de las diferentes actividades tanto académicas, como administrativas que se realizan dentro de las escuelas y las Facultades, así como del control de usuarios y servicios que se les proporcionan.

La plataforma COMA presta servicios que están orientados a promover la interacción entre los miembros de la comunidad. Actualmente el grupo Calumet brinda soporte al servicio Trabajos de Grado, en el marco de este proceso se hace necesario realizar cambios, modificaciones y nuevas implementaciones tales como:

Permitir a los directores solicitar ante el Comité de Proyectos de Grado retirarse de un proyecto, guardar la fecha y comentario del evaluador de solicitud al emitir el concepto de la solicitud.

En el portal de grupos se hace necesario realizar cambios, modificaciones y nuevas implementaciones tales como: Nueva página de entrada con la posibilidad de publicar banner, texto explicativo, enlace y soporte al portal de grupos para eventos.

Actualmente no existe un módulo en el portal de grupos para realizar inscripciones, por lo tanto, se hace necesario agregar un módulo de inscripciones para los diferentes roles con la posibilidad de subir archivos. Además, la posibilidad que los inscritos a estas, sean agregados como integrantes del grupo.

---

\* Trabajo de grado

\*\* Facultad de Ingenierías Físico-mecánicas. Escuela de Ingeniería de Sistemas e Informática. Director: Msc. Luis Ignacio González Ramírez

## ABSTRACT

**TITLE:** Analysis, maintenance, design, development and implementation of new functionalities and improvements related to the modules of the Group Web Portal and of the Degree Projects in the COMA platform Academic Community. \*

**AUTHOR:** Alexy Yessenia Rincón Capera\*\*

**KEY WORDS:** Portal, Groups, Jobs.

## DESCRIPTION

At present the schools have a web-oriented information system that is in charge of the administration and control of the different academic and administrative activities that are carried out within the schools and the Faculties, as well as the control of users and services that are provided.

The COMA platform provides services that are oriented to promote interaction among community members. Currently, the Calumet group provides support to the Grade Works service, within the framework of this process it is necessary to make changes, modifications and new implementations such as:

Allow the directors to request before the Graduate Project Committee to withdraw from a project, save the date and comment of the request evaluator when issuing the concept of the application.

In the group portal it is necessary to make changes, modifications and new implementations such as: New entry page with the possibility of publishing banner, explanatory text, link and support to the group portal for events.

Currently there is no module in the group portal to make registrations, therefore, it is necessary to add an enrollment module for the different roles with the possibility of uploading files. In addition the possibility that those registered to these, are added as members of the group.

---

\* Degree Work

\*\* Faculty of Physico-Mechanical Engineering. Department of Systems Engineering and Computing Science. Supervisor Msc. Luis Ignacio González Ramírez

## INTRODUCCIÓN

Los Portales Web Comunidad Académica son los principales canales de comunicación e integración entre los miembros de la comunidad de las escuelas y miembros en general de la comunidad UIS. Actualmente los servicios se han extendido a más escuelas de las distintas facultades gracias a la aceptación de los usuarios por lo que se adopta el objetivo de mantener y mejorar los servicios que se ofrecen, además de crear nuevos servicios que satisfagan las necesidades crecientes de los usuarios de la comunidad académica.

El grupo CALUMET, grupo de desarrollo de software de la escuela de Ingeniería de Sistemas, se encarga de desarrollar los nuevos servicios y darle mantenimiento a los existentes de manera que su actualización responda al continuo cambio. Con el fin de llevar a su cumplimiento el objetivo principal del grupo y proporcionar portales web con contenido dinámico, se cuenta con herramientas software de libre distribución como Netbeans, Github, Sqlyog, JavaScript, jQuery, entre otras.

En este documento se presenta un soporte teórico, metodológico y técnico del desarrollo realizado en el servicio de Trabajos de Grado y Portal Web de Grupos con el fin de soportar las necesidades de la comunidad se realizaron los siguientes cambios, modificaciones y nuevas implementaciones. La creación de un nuevo servicio de intercambio de mensajes que permita la comunicación entre autores y director en el módulo trabajos de Grado; Implementación de un nuevo servicio que permita a los directores solicitar ante el comité de Trabajos de Grado retirarse de un proyecto y por último reingeniería a la página de entrada y agregar el módulo de inscripciones del portal de grupos.

## **1. PLANTEAMIENTO DEL PROYECTO**

### **1.1 DEFINICIÓN DE LA SITUACIÓN ACTUAL**

En la actualidad las Escuelas y las Facultades de la UIS, cuentan con un sistema de información orientado a la Web que se encarga de la administración y control de las diferentes actividades tanto académicas como administrativas que se realizan dentro de la escuela y la Facultad, así como del control de usuarios y servicios que se les proporcionan.

Los servicios de los portales Web de las escuelas deben mejorar constantemente y adaptarse a los cambios que se presenten en su entorno, a su vez deben dar solución a los problemas y necesidades que surjan por parte de los usuarios del sistema para incrementar su tiempo de vida útil y no llegar a convertirse en un software obsoleto, razón por la cual las labores de mantenimiento y actualización se hacen indispensables.

Actualmente en el módulo Trabajos de Grado no existe un espacio de comunicación que permita la interacción entre autores y director en la sección de Subir Plan, para esto se debe generar el espacio de comunicación en dicha sección. No existe actualmente un servicio que permita al director retirarse de un proyecto por lo cual se realiza este nuevo servicio con el fin de que el director solicite al comité de trabajos de grado retirarse de un proyecto. La página de entrada del portal de grupos se encuentra desactualizada, luego se hace necesario modificar dicha página con el fin de hacerla más amigable a los diferentes usuarios. Hoy en día no existe un módulo de inscripciones en el portal de grupo, por tanto, se hace necesario realizar este nuevo servicio para los diferentes roles con la posibilidad de subir archivos y agregar inscritos como miembros del grupo para que puedan ingresar e interactuar.

## 1.2 JUSTIFICACIÓN

El Portal Web Comunidad Académica (COMA), que desarrolla y mantiene el grupo Calumet, es una plataforma que provee lugares, adicionales a los ya existentes, de encuentro y comunicación a la comunidad de los programas de las diferentes escuelas. Esta plataforma contribuye a la realización de la misión en sus tres ejes: extensión, investigación y docencia. En ella, la Comunidad podrá mantener actualizada su hoja de vida, publicar noticias, proponer eventos, consultar y crear agendas, compartir documentos, enviar correos, y otros servicios que podrán hacer más fácil la realización de la misión institucional. Los servicios de los portales web deben actualizarse constantemente, mejorar y adaptarse a los cambios que se presenten en su entorno, satisfaciendo las necesidades de los usuarios del sistema. Esto permite aumentar el tiempo de vida útil del software evitando que pase a ser obsoleto con la creación de nuevos servicios, y con el mantenimiento realizado a los existentes.

A solicitud de los Grupos de las Escuelas se hace necesario actualizar el Portal de Grupos. Esta nueva versión debe incluir:

- Nueva página de entrada con posibilidad de publicar banner, texto explicativo y enlace como en el Portal de las Escuelas.
- Soporte al Portal de Grupos para eventos con funcionalidades como: contador y estadísticas de dispositivo, país y ciudad, sistema operativo, y otros datos sobre los visitantes.
- Crear el módulo de inscripciones públicas, escuela y UIS para las diferentes necesidades del grupo como talleres, seminarios, congresos, eventos, nuevos miembros y otros. Además, se debe tener la posibilidad de agregar inscritos como miembros del grupo para ingresar e interactuar.

En el módulo de Trabajos de Grado se deben realizar cambios y modificaciones solicitadas por las Escuelas:

- Servicio para permitir a los Directores solicitar ante el Comité de Trabajos de Grado retirarse de un proyecto. Debe hacerse todo el soporte para que sea visto por el comité en sesión, comité en sesión individual, mostrarse de manera correcta en el acta, mostrarse de manera correcta en la información del proyecto, debe permitir a los miembros del comité en la sesión virtual tomar una posición y hacer un comentario. Además, debe permitir al director retirarse, es decir, retirar la solicitud hasta antes de la sesión del comité.
- Guardar fecha y comentario del evaluador de solicitud al emitir concepto de la solicitud. Además, debe enviar correo a los autores y director de proyecto.
- Capturar y guardar el comentario del director a los autores al revisar el plan. Este mensaje debe enviarse a los autores en un correo.

### **1.3 OBJETIVOS**

**1.3.1 Objetivo general** Realizar las funciones de mantenimiento, análisis, diseño, desarrollo e implementación de nuevos servicios y página de ingreso para el Portal Web de Grupos, y de nuevas funcionalidades del módulo de Trabajos de Grado con el fin de hacerlos óptimos, eficientes y sostenibles, permitiendo acceder a la información de manera más sencilla, ágil, óptima y eficiente.

#### **1.3.2 Objetivos específicos**

**1.3.2.1.** Realizar una actualización al Portal Web de Grupos que permita nuevas funcionalidades en la página de entrada como publicar banner, texto explicativo y redes sociales.

**1.3.2.2.** Agregar al Portal de Grupos funcionalidades tales como contador y estadísticas de dispositivo, país y ciudad, sistema operativo, y otros datos sobre los visitantes.

**1.3.2.3.** Crear el módulo de inscripciones públicas, escuela y UIS para las diferentes necesidades del grupo como talleres, seminarios, congresos, eventos, nuevos miembros y otros. Este módulo debe permitir hacer seguimiento, intercambiar documentos y posibilidad de enviar correos entre los interesados inscritos y el administrador de la inscripción.

**1.3.2.4.** Posibilidad de agregar inscritos como miembros del grupo para ingresar e interactuar. Esta labor la debe realizar solo los administradores de grupo.

- 1.3.2.5.** En el módulo de Trabajos de Grado debe permitir al director solicitar ante el Comité de Trabajos de Grado su retiro como director de un proyecto. Debe hacerse todo el soporte para que sea visto por el comité en sesión, comité en sesión individual, mostrarse en el acta, mostrarse de manera correcta en la información del proyecto. Debe permitir a los miembros del comité en la sesión virtual tomar una posición y hacer un comentario. Además, permitir al director retractarse, es decir, retirar la solicitud hasta antes de la sesión del comité.
- 1.3.2.6.** Guardar fecha y comentario del evaluador de solicitud al emitir concepto y almacenarlo de tal forma que se muestren todos los comentarios emitidos anteriormente al comité y al evaluador. Además, debe enviar correo a los autores y director de proyecto.
- 1.3.2.7.** Capturar y guardar el comentario realizado por el director dirigido a los autores al revisar el plan. Este mensaje debe enviarse a los autores en un correo.

## **1.4 IMPACTO Y VIABILIDAD**

**1.4.1 Impacto.** Los portales web de las escuelas han sido una herramienta útil para el manejo de la información, por lo tanto, es necesario realizar labores de administración y mantenimiento, para ofrecer al usuario un sitio más confiable.

Se pretende que los procesos que se realizan en las escuelas cada día sean más ágiles, dinámicos, seguros y eficientes, permitiendo una mejor organización de la información, razón por la cual se crean nuevos servicios y se hace reingeniería a servicios existentes para que se ajusten a las necesidades que puedan surgir.

**1.4.2 Viabilidad.** La administración del sitio soporte a usuarios, mantenimiento y desarrollo de nuevos servicios es completamente viable pues se usará software de libre distribución, recurso humano preparado para tal fin, servidores que marchan de manera legal en la escuela y la facultad, equipos disponibles y todo el soporte tecnológico necesario para el desarrollo del mismo.

Además, se cuenta con la supervisión por parte del director del proyecto y la colaboración del equipo de trabajo CALUMET, agentes de gran apoyo en la realización de este proyecto.

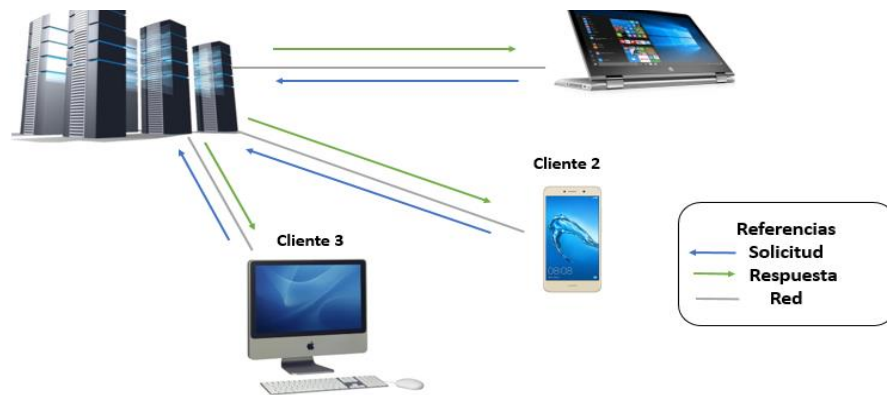
## **2. MARCO TEÓRICO**

### **2.1 ARQUITECTURA CLIENTE/SERVIDOR**

En la arquitectura cliente/servidor cada uno de los clientes produce un mensaje solicitando un determinado servicio a un servidor (hace una petición) estos envían uno o varios mensajes como respuesta (responden peticiones o provee un servicio). La mayoría del trabajo pesado (procesos de base de datos, procesar la lógica de la aplicación entre otros) está a cargo de los servidores, el cliente comúnmente se encarga de las funciones de administración de la interfaz de usuario, interacción con el usuario, recibir resultados del servidor, generar requerimientos de base de datos, entre otros.

Esta idea se puede aplicar tanto a programas que se están ejecutando en una sola máquina, pero es más ventajosa en un sistema operativo multiusuarios distribuidos a través de una red de computadores.

Figura 1. Modelo Cliente/Servidor



**2.1.1 Características de la arquitectura Cliente/Servidor.** Las características básicas de una arquitectura Cliente / Servidor son:

- Es quien inicia solicitudes o peticiones, tienen por tanto un papel activo en la comunicación.
- El proceso del cliente da la interface entre usuarios y el resto del sistema, maneja recursos compartidos tales como bases de datos, impresoras, módems, etc.
- El cliente y el servidor pueden actuar como una sola entidad y también pueden actuar como entidades separadas, realizando actividades independientes.
- Las tareas del cliente y el servidor tienen diferentes requerimientos como: velocidad del procesador, memoria o capacidad del disco, por tanto, la plataforma de hardware y el sistema operativo del cliente y del servidor no son siempre la misma y eso se conoce como ambiente heterogéneo.
- La escalabilidad horizontal permite agregar más estaciones de trabajo activas sin afectar el rendimiento y la escalabilidad vertical permite mejorar las características del servidor o agregar múltiples servidores. Se puede realizar independientemente cambios en las plataformas de los clientes o de los servidores, ya sea actualización o reemplazo tecnológico, de manera transparente para el usuario final.

**2.1.2 Clasificación de las arquitecturas Cliente/Servidor.** Los sistemas Cliente/Servidor se clasifican de acuerdo al nivel de abstracción del servicio que se ofrece. Se distinguen tres componentes básicos de software para la clasificación:

- **Presentación:** Presentación de resultados al usuario de forma comprensible.
- **Lógica de aplicación:** Esta capa es la responsable del procesamiento de la información que tiene lugar en la aplicación.
- **Base de datos:** Está compuesta por los archivos que contienen los datos persistentes de la aplicación.

La siguiente es la clasificación de la arquitectura Cliente/Servidor

**2.1.2.1 Arquitectura Cliente/Servidor de dos capas.** Consiste en una capa de presentación y lógica de la aplicación; y otra de la base de datos, cuando el cliente solicita recursos entonces el servidor responde directamente a la solicitud con sus propios recursos. Normalmente esta arquitectura es utilizada en las siguientes situaciones:

- Cuando se requiere poco procesamiento de datos en la organización.
- Cuando se tiene una base de datos centralizada en un solo servidor.
- Cuando la base de datos es relativamente estática.
- Cuando se requiere un mantenimiento mínimo.

**2.1.2.2 Arquitectura Cliente/Servidor de tres capas.** Define como organizar el modelo de diseño en capas, que pueden estar físicamente distribuidas, es decir que los componentes de una capa solo pueden hacer referencia a componentes en capas inferiores. Este patrón es importante porque simplifica la comprensión y la organización del desarrollo de sistemas complejos, reduciendo las dependencias de forma que las capas más bajas no conscientes de ningún detalle o interfaz de las superiores, está compuesta de:

- Un equipo cliente con una interfaz de usuario (habitualmente se utiliza un navegador web), que solicita los recursos.
- El servidor de aplicaciones (o software intermedio), cuya tarea es prestar los recursos solicitados, pero que requiere de otro servidor para hacerlo.
- El servidor de datos que almacena y proporciona al servidor de aplicaciones los datos que requiere.

**2.1.3 Arquitectura Cliente/Servidor aplicada.** En el desarrollo de este proyecto se recurre a arquitectura de tres capas, debido a las ventajas ofrecidas como: Escalabilidad, fácil mantenimiento y el manejo de un mayor número de usuarios que la ofrecida por la arquitectura C/S de dos capas. La arquitectura es aplicada de la siguiente forma:

- Capa de Cliente: Interfaz con el usuario, se usa un navegador web.
- Capa Intermedia: Para los servicios del negocio se utiliza un computador configurado como servidor web, el cual almacena el portal web conformado por páginas JSP y JavaBeans. Allí se realizan los procesos complejos, y se solicitan los servicios del servidor de datos cuando es necesario acceder a la información almacenada en la base de datos.
- Capa de Servidor: Se utiliza el motor de bases de datos MySQL, el cual se encuentra en el mismo servidor web.

#### **2.1.4 Ventajas del esquema Cliente/Servidor**

- La existencia de plataformas de software y hardware de varios fabricantes y cada vez más a económicas contribuye a la reducción de costos y favorece la flexibilidad en la implantación y actualización de soluciones.
- Este esquema facilita la integración entre sistemas heterogéneos y comparte información permitiendo que las máquinas existentes puedan ser utilizadas con interfaces amigables al usuario, de esta forma integrar los computadores con sistemas medianos y grandes, sin necesidad de que todos tengan que utilizar el mismo sistema operacional.
- Facilita a los diferentes departamentos de una organización soluciones locales, permitiendo la integración de la información principal totalmente.

#### **2.1.5 Desventajas del esquema Cliente/Servidor**

- El mantenimiento de los sistemas es complejo pues implica la interacción de diferentes partes hardware y software de diferentes proveedores, lo cual dificulta el diagnóstico de fallas.
- Se cuenta con escasas herramientas para la administración y ajuste del desempeño de los sistemas, además se deben tener estrategias para el manejo de errores y para salvaguardar la consistencia de los datos.
- La seguridad del esquema C/S es preocupante, un ejemplo: las validaciones y verificaciones que se deben hacer tanto en el cliente como en el servidor.
- El desempeño es un aspecto a tener en cuenta en el esquema C/S, problemas de este estilo pueden presentarse por congestión en la red.

## **2.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS**

Las páginas dinámicas aportan grandes beneficios porque permiten entrar a bases de datos para extraer información que pueda presentarse al usuario, dependiendo de algunos permisos y de la misma forma para almacenar información.

Existen diferentes tecnologías para el desarrollo de páginas dinámicas entre ellas están:

**2.2.1 Código del Lado del Cliente (Client Side Scripts).** Código ejecutado por los navegadores, el cual los computadores clientes tienen instalados. Las tecnologías más comunes de este tipo son:

- JavaScript: Lenguaje de programación interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C. Permite, crear ventanas, mostrar texto en movimiento y verificar las entradas a un formulario.
- Controles Activos: tecnología Microsoft que permite la creación de aplicaciones Windows, como pueden ser Visual Basic Script o Visual C. Es la respuesta de Microsoft a los Applets de Java.
- Java Applets: Programas escritos en lenguaje de programación Java, se incrustan en HTML y se ejecutan en el navegador gracias a la Máquina Virtual de Java (JVM) que lleva éste incorporado.

**2.2.2 Código del Lado del Servidor (Server Side Scripts).** Código que se ejecuta en el servidor. Para su actividad el programa ejecuta y procesa los datos o peticiones que el usuario envía desde su navegador, para luego enviar los resultados del programa en una página HTML que el usuario verá normalmente en su navegador. Los más usados son:

- ASP (Active Server Pages): Permite crear dinámicamente páginas Web mediante HTML, scripts, y componentes de servidor ActiveX reutilizables,

requiere de un computador configurado como Servidor Web de Microsoft (Microsoft Web Server), el navegador del cliente es indiferente pues el trabajo se realiza del lado del servidor. Da gran uso en la gestión de Bases de Datos ya que puede conectarse a SQL, Access, Oracle u otras.

- PHP (PHP Hypertext Pre-processor): Lenguaje de programación interpretado, diseñado para la creación de páginas web dinámicas. Es un lenguaje de código abierto (Open Source) y gratuito. Su gran potencia se encuentra en la interacción con los motores de bases de datos como Oracle y MySQL.
- JSP (Java Server Pages): tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo. Permiten la utilización de código Java mediante scripts.

**2.2.3 Tecnología aplicada.** La tecnología aplicada para la creación del portal web fue JSP, por lo tanto, los nuevos servicios son desarrollados con esta misma tecnología, ya que permite producir aplicaciones independientes de la plataforma y portables a otros sistemas operativos y servidores web.

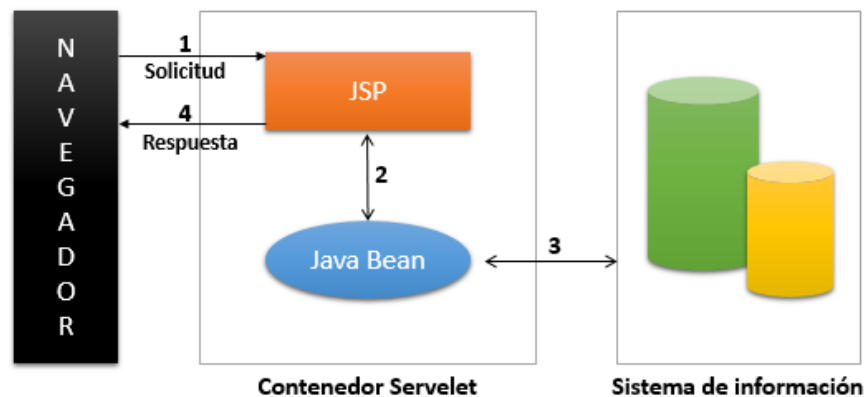
Las paginas JSP y servlets se ejecutan en la Máquina Virtual de Java, lo cual permite que se puedan usar en cualquier tipo de computador, siempre y cuando este instalada la Máquina Virtual de Java. Cada JSP se ejecuta en su propio contexto (llamado también hilo o hebra); pero no se comienza a ejecutar cada vez que recibe una petición, sino que persiste de una petición a la siguiente, de forma que no se pierde tiempo en invocarlo. Su persistencia permite hacer cosas de forma más eficiente como la conexión a bases de datos y manejo de sesiones.

Una página JSP se compila a una aplicación Java la primera vez que se invoca, y de esta aplicación Java se crea una clase que empieza a ejecutarse en el servidor como un servlet. Un JSP es una página web con etiquetas especiales y código Java incrustado, mientras que un servlet es un programa que recibe peticiones y genera a partir de ellas una página web.

### 2.2.3.1 Modelo de acceso a JSP

- Un usuario en su navegador web cliente hace una petición que es enviada a un archivo JSP. Este archivo accede a componentes del servidor que generan contenido dinámico y lo presentan en el navegador.
- Después de recibir la petición del cliente, el archivo JSP pide información de un JavaBean si es necesario.
- El JavaBean en turnos puede pedir información de otro JavaBean o de una base de datos.
- Una vez el JavaBean genera el contenido, el archivo JSP puede consultar y presentar el contenido del JavaBean al navegador.

*Figura 2. Modelo de acceso a JSP*



La primera vez que un archivo JSP es invocado, este es compilado en un objeto, la respuesta del objeto es HTML estándar, el cual es interpretado por el navegador para ser presentado al usuario. Después de la compilación, el objeto de la página es almacenado en la memoria del servidor. En las peticiones posteriores a esta página, el servidor revisa si el archivo JSP ha cambiado. Si no ha cambiado, el

servidor utiliza el objeto de la página compilada guardado en memoria para generar la respuesta al cliente, en caso contrario el servidor automáticamente compila el archivo de la página y reemplaza el objeto en la memoria.

## **2.3 BASES DE DATOS**

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso, con una redundancia controlada y una estructura que refleja las interrelaciones y restricciones existentes en el mundo real. En la base de datos se almacena información considerada necesaria para una determinada organización o negocio.

Existen modelos que describen la estructura de una base de datos (entidades, atributos y relaciones), la mayoría de los modelos de datos poseen un conjunto de operaciones básicas como consultar y actualizar y eliminar.

**2.3.1 Modelos de Bases de Datos.** Las bases de datos se clasifican de acuerdo con su modelo de administración de datos. Algunos modelos utilizados con frecuencia son:

**2.3.1.1 Base de Datos Jerárquica.** Estas bases de datos almacenan su información en una estructura escalonada, organizando los datos en forma similar a un árbol (visto al revés), en donde un nodo padre de información puede tener varios hijos, el nodo que no tiene padres es llamado raíz, y a los nodos que no tienen hijos se les conoce como hojas. Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento. Esta limitado por su incapacidad de representar eficientemente la redundancia de datos.

**2.3.1.2 Base de Datos de Red.** En este modelo se permite que un mismo nodo tenga varios padres. Ofrece una solución eficiente al problema de redundancia de datos; sin embargo, la dificultad para administrar los datos en una base de datos de red ha conllevado a que sea un modelo usado más por programadores que por usuarios finales.

**2.3.1.3 Base de Datos Relacional.** Es el más utilizado para modelar problemas reales y administrar datos dinámicamente. Su fundamento es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos, también llamados tuplas. Cada relación es una tabla que está compuesta por registros (las filas de una tabla), que representan las tuplas, y campos (las columnas de una tabla). Los datos pueden ser recuperados o almacenados mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información. El lenguaje más habitual para construir las consultas a bases de datos relacionales es el Lenguaje Estructurado de Consultas (Structured Query Language, SQL), un estándar implementado por los principales manejadores de bases de datos relacionales.

**2.3.2 Manejadores o Gestores de Bases de Datos.** El sistema manejador de bases de datos es la porción más importante del software de un sistema de base de datos. Un DBMS es una colección de numerosas rutinas de software interrelacionadas, cada una de las cuales es responsable de alguna tarea específica.

Las funciones principales de un DBMS son:

- Crear y organizar la Base de Datos.
- Establecer y mantener las trayectorias de acceso a la base de datos de tal forma que los datos puedan ser capturados rápidamente.
- Manejar los datos de acuerdo a las peticiones de los usuarios.

- Registrar el uso de las bases de datos.
- Interacción con el manejador de archivos a través de las sentencias en Lenguaje Manipulador de Datos (Data Manipulation Language, DML) al comando del sistema de archivos.
- Respaldo y recuperación: Consiste en contar con mecanismos implantados que permitan la recuperación fácilmente de los datos en caso de ocurrir fallas en el sistema de base de datos.
- Control de concurrencia: consiste en controlar la interacción entre los usuarios concurrentes para preservar la consistencia de los datos.
- Seguridad e Integridad: consiste en contar con mecanismos que permitan el control de la consistencia de los datos evitando que estos se vean perjudicados por cambios no autorizados o previstos.

**2.3.3 MySQL.** Es un sistema de base de datos operacional considerado uno de los más importantes, utilizado por usuarios del medio para el diseño y programación de base de datos de tipo relacional. MySQL se usa como servidor a través del cual pueden conectarse múltiples usuarios y utilizarlo al mismo tiempo. La característica más interesante de MySQL es que permite recurrir a las bases de datos multiusuario a través de la web y en diferentes lenguajes de programación y diferentes plataformas que se adaptan a diferentes necesidades y requerimientos, además MySQL es conocida por desarrollar alta velocidad de búsqueda de datos e información, a diferencia de sistemas anteriores.

#### **2.3.4 Ventajas de MySQL**

- El MySQL es un Open Source, o sea código abierto que puede ser usado y modificado.
- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.

- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consume puede ser ejecutado en una maquina con escasos recursos sin ningún problema.
- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
- Su conectividad, velocidad, y seguridad hacen de MySQL altamente apropiado para acceder a bases de datos en internet.

## **2.4 NETBEANS**

Es un entorno de desarrollo integrado (IDE), siendo una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas escritos en JAVA, pero puede servir para cualquier otro tipo lenguaje de programación. Netbeans es un producto libre y gratuito sin restricciones de uso.

- El Netbeans es un entorno de desarrollo integrado de código abierto escrito completamente en Java usando la plataforma Netbeans, soporta desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles).
- La versión actual es NetBeans IDE 8.1 Desde NetBeans IDE 6.5 se extienden las características existentes del Java EE (incluyendo Soporte a Persistencia, EEJB 3 y JAX-WS). Adicionalmente, el Netbeans Enterprise Pack soporta el desarrollo de Aplicaciones empresariales java EE 5, incluyendo herramientas de desarrollo visuales de SOA, herramientas de esquemas XML, orientación a web servicios (for BPEL), y modelado UML.
- Todas las funciones del IDE son provistas por módulos. Cada Módulo provee una función bien definida, tales como el soporte de Java, edición, o soporte para el sistema de control de versiones. Netbeans contiene todos los módulos necesarios para el desarrollo de aplicaciones Java en una sola descarga, permite al usuario comenzar a trabajar inmediatamente.

## **2.5 SISTEMA DE CONTROL DE VERSIONES**

Un sistema de control de versiones es un software que administra el acceso a un conjunto de ficheros, y mantiene un historial de cambios realizados. El control de versiones es útil para guardar cualquier documento que cambie con frecuencia, o el código fuente de un programa.

Normalmente consiste en una copia maestra en un repositorio central, y un programa cliente con el que cada usuario sincroniza su copia local. Además, el repositorio guarda registro de los cambios realizados por cada usuario, y permite volver a un estado anterior en caso de necesidad.

Existen multitud de sistemas de control de versiones, pero sin duda, el más popular es CVS (Concurrent Versions System). CVS tuvo el mérito de ser el primer sistema usado por el movimiento de código abierto para que los programadores colaboran remotamente mediante el envío de parches. Es de uso gratuito, código abierto, y emplea fusión de cambios. Subversión se creó para igualar y mejorar la funcionalidad de CVS, preservando su filosofía de desarrollo.

### **2.5.1 Subversión.**

Sistema de control de versiones iniciado por CollabNet Inc. Emplea licencia Apache/BSD. Se usa para mantener versiones actuales e históricas y los cambios de archivos tales como los de código fuente, páginas web y/o documentación. Esto permite recuperar versiones antiguas de los datos o examinar cómo han ido evolucionando esto. Su objetivo es ser un sucesor prácticamente compatible del ampliamente usado Concurrent Version system (CVS).

Subversión puede trabajar a través de redes, lo que permite que las personas que estén en diferentes computadores puedan usarlo, con la posibilidad de que varias personas modifiquen y gestionen el mismo conjunto de datos desde sus sitios promueve la colaboración, y como el trabajo está versionado, ya que, si se produce algún cambio incorrecto de los datos, sólo hace falta deshacerlo.

## 2.6 PROGRAMACIÓN UTILIZADA

Para el desarrollo de este proyecto se usó la Programación Orientada a Objetos (P.O.O.). La P.O.O. es una de las formas más populares de programas que usa objetos y sus interacciones para diseñar aplicaciones y programas de computador, intenta simular el mundo real a través del significado de objetos que contienen características y funciones; abstrae algunas características de sistemas naturales complejos como son:

- Atributos: estado del objeto.
- Métodos: comportamiento del objeto.
- Herencia: comportamientos comunes entre objetos relacionados para hallar relaciones de especialización y generalización de comportamientos.

**2.6.1 Clases.** Definición de todos los elementos que componen un objeto. Cuando se programa un objeto y se definen sus características y funcionalidades, realmente se programa una clase. Por lo tanto, para realizar la abstracción de sistemas naturales, observamos y analizamos un grupo de cosas con características comunes, el resultado de esta abstracción será válido para todas estas cosas.

**2.6.2 Objetos.** Cualquier cosa real o abstracta, que posee atributos y un conjunto de operaciones que manipulan esos atributos que da un comportamiento particular. Un objeto es una instancia de una clase, el estado del objeto se determina por el estado (valor) de sus propiedades o características (atributos).

**2.6.3 Atributos.** Características de un objeto siendo un conjunto de datos (valores) y calificadores para aquellos datos. Estos atributos pueden ser desde tipos de datos simples (enteros, caracteres, cadenas de texto) hasta otros objetos.

**2.6.4 Métodos.** Son funciones o procedimientos propios de la clase que pueden tener acceso a los atributos de la misma para realizar las operaciones para los que son programados.

**2.6.5 Herencia.** Se fundamenta en usar una clase ya creada para tomar sus características en clases más especializadas o derivadas de ésta para reutilizar el código que sea común con la clase base, y solamente definir nuevos métodos o redefinir algunos de los existentes para ajustarse al comportamiento particular de esta subclase.

#### **2.6.6 Beneficios de la Programación Orientada a Objetos**

- Permite obtener aplicaciones modificables y fácilmente extensibles a partir de componentes reutilizables.
- Disminución en el tiempo de desarrollo gracias a la reutilización del código.
- El desarrollo del software es más intuitivo porque las personas piensan naturalmente en términos de objetos más que en términos de algoritmos de software.

A continuación, se presenta una breve descripción de Java, el lenguaje de programación orientado a objetos que se usó en el desarrollo de este proyecto:

**2.6.7 Java y JDK (Java Development Kit).** Java es un lenguaje desarrollado por Sun Microsystems, en el año 2009 fue adquirida por la compañía Oracle.

Permite escribir aplicaciones que puedan ejecutarse en casi cualquier plataforma. El lenguaje toma parte de la sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria. Además, cuenta con una característica denominada “recolección de basura”, que examina la memoria y

libera cualquier variable u objeto que no esté siendo usado. El JDK es un software que provee herramientas de desarrollo para la creación de programas en java.

Para trabajar con Java se necesita un kit de desarrollo que proporciona:

- Un compilador: javac
- Un intérprete: java.
- Un generador de documentación: javadoc
- Un visor de applet para generar sus vistas previas, ya que un applet carece de método main y no se puede ejecutar con el programa java: Appletviewer.

## **2.7 SERVIDORES WEB**

Es un tipo de software que se encuentra a la espera de una petición hecha por una aplicación cliente y da respuesta a dicha petición a través de una página web. Para cada transacción el servidor debe realizar dos acciones básicas: integrar todos los componentes de la página (texto, imágenes, vídeo, scripts, etc.) y enviarla rápidamente al usuario. A continuación, se describe el servidor Web que se ajusta a la tecnología escogida para el proyecto.

**2.7.1 Servidor Apache Tomcat.** Servidor de aplicaciones Java basado en los estándares definidos por Sun Microsystems. Tomcat es desarrollado como parte del proyecto de código abierto Jakarta de la fundación de software Apache y es uno de los servidores de aplicaciones Java más utilizados, en especial porque es liviano, cumple con todos los estándares, sencillo de instalar, tiene muy buena documentación y es gratuito, además por ser escrito en Java funciona en cualquier sistema operativo que disponga de la Máquina Virtual de Java (JVM).

Es posible ejecutarlo desde la línea de comandos (consola o terminal), después de configurar algunas variables de entorno, sin embargo, configurar cada variable de entorno y seguir los parámetros de la líneas de comando usados por Tomcat es

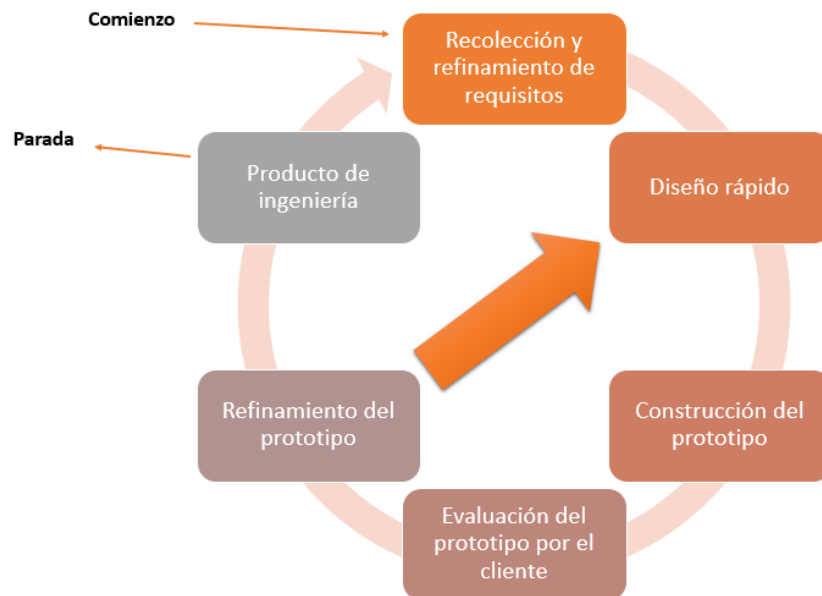
tedioso y expuesto a errores, en su lugar se proporciona código existente para arrancar y detener el servicio.

### 3. MARCO METODOLÓGICO

#### 3.1 PROTOTIPO EVOLUTIVO

Para realizar los nuevos servicios para los portales web comunidad académica de las diferentes escuelas y facultades de la Universidad Industrial de Santander se propone como metodología de desarrollo el prototipo evolutivo.

*Figura 3. Prototipo evolutivo*



La elección de la metodología de prototipo evolutivo se debe a las siguientes razones:

- Es deseable tener un bosquejo de lo que se desee mejorar o crear para poder incorporar sugerencias de cambio por parte de los usuarios del portal de las escuelas en las etapas tempranas del desarrollo.

- Por otra parte, es necesario saber lo antes posible si hemos interpretado correctamente las especificaciones y las necesidades de las escuelas y de los profesores.
- En muchos casos los usuarios no tienen una idea definida de lo que desean, por lo tanto, debemos tomar decisiones y suponer qué es lo que el usuario quiere.
- Por este motivo, la emisión de los prototipos brinda la posibilidad de efectuar refinamientos de los requerimientos en forma sucesiva a fin de acercarse al producto deseado. Con el prototipo evolutivo se comienza diseñando y construyendo las partes más importantes de la aplicación en un prototipo que posteriormente se refinara y ampliará hasta que el prototipo se termine. Este prototipo será el software que se entregará al final.
- La decisión se fundamenta en la ventaja de la realización de los cambios en etapas tempranas y la posibilidad de emisión de varios prototipos evaluables durante el desarrollo, obteniéndose de este modo, y de forma paralela, una metodología integral también para el proceso de evaluación del programa.
- Esta metodología propicia un intercambio de conocimientos y de autocrítica al sistema, lo que conlleva a que se produzcan muchas pruebas antes de liberar una nueva versión, así como mejoras rápidas a problemas que puedan surgir durante su uso.

Procedimiento a seguir para la metodología planteada:

- Hacer un análisis de los requerimientos para la construcción de los prototipos.
- Desglosar los objetivos globales con el fin de tener una idea más detallada del software a realizar, mediante reuniones entre los desarrolladores y los usuarios, en las cuales se identifican los requerimientos de los usuarios y se concluyen los aspectos que requieren una mayor definición.
- Presentar al usuario el diseño de un prototipo enfocado en los aspectos visuales del software, métodos de entrada y formatos de salida, para proceder a la construcción del mismo.

- Evaluación del prototipo por parte del usuario para filtrar los requisitos del software a desarrollar.
- Se produce un proceso interactivo en el que el prototipo es depurado para satisfacer necesidades del usuario, de igual forma el desarrollador obtiene una mejor comprensión de lo que hay que hacer para la entrega del producto final de ingeniería requerido por el usuario.

## **3.2 LENGUAJE DE MODELADO UNIFICADO**

El Lenguaje de Modelado Unificado o Unified Modeling Language (UML), es el más utilizado en la actualidad. Es un lenguaje gráfico estándar para visualizar, especificar, construir y documentar un sistema para describir un modelo del sistema, incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

UML no es un método de desarrollo porque no indica los pasos a seguir para llegar al código, es decir, no especifica como pasar del análisis al diseño y de este al código. Al no ser un método de desarrollo resulta ser independiente del ciclo de desarrollo que se siga, puede encajar en un ciclo en cascada, evolutivo, espiral o en métodos ágiles de desarrollo.

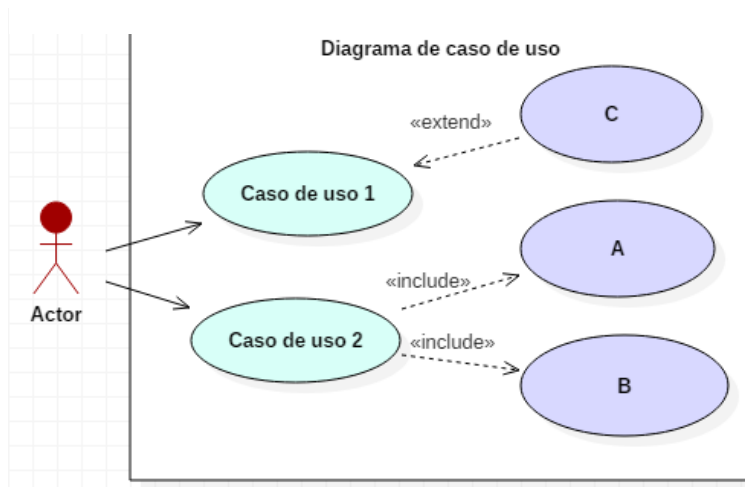
**3.2.1 Diagramas de UML.** Los diagramas UML utilizados en el desarrollo de este proyecto fueron diagramas de casos de uso y diagramas de secuencias. Las principales razones por las cuales se prefirió UML como el lenguaje de modelado son:

- UML tiene una notación gráfica muy expresiva que permite representar todas las fases de un proyecto informático, desde el análisis con casos de uso, el diseño con diagramas de clases, objetos, etc.

- UML facilita el entendimiento de la información, la función y el comportamiento de un sistema, haciendo fácil el análisis de los requerimientos, ya que sirve de apoyo en los procesos de análisis de un problema.
- UML permite a los creadores de sistemas realizar diseños que faciliten la comunicación a otras personas de manera convencional.
- UML permite generar un punto de comparación entre lo logrado y lo planificado.

**3.2.2 Diagramas de casos de uso.** Representación gráfica del entorno del sistema (actores) y su funcionalidad principal. Describe lo que hace el sistema desde el punto de vista de un observador externo, concentrándose en expresar lo que hace el sistema y no en dar respuesta de cómo lograr su comportamiento.

*Figura 4. Diagramas de casos de uso.*



**Actores:** Un actor en un caso de uso representa un rol, que alguien o algo puede desempeñar dentro un sistema y no un alguien o algo específico.

En este proyecto se destacan tres clases de actores:

- **Administradores:** Son usuarios que además de pertenecer a la categoría de usuarios tienen un perfil de administrador, con el cual tiene permisos extras a los que tiene un usuario comúnmente dentro del sitio; alguno de estos son los auxiliares de administración del portal, los profesores, las secretarías de las escuelas, entre otras. Dentro de esta categoría se incluye también el súper administrador.
- **Súper Administrador:** Es el usuario que puede administrar, controlar y modificar los portales web de las escuelas, sus parámetros y sus usuarios.
- **Usuario Portal Web Comunidad Académica:** Es el tipo de usuario común de los portales web y a quien van dirigidas las páginas de servicio. Este usuario solo tiene control sobre sus servicios permitidos.

**Inclusión (include–uses):** Es una forma de interacción, un caso de uso dado puede "incluir" otro. Una inclusión es utilizada para indicar que un caso de uso depende de otro, es decir, la funcionalidad de determinado caso de uso se requiere para realizar las tareas de otro. En la figura 4 el caso de uso "Caso de uso 2" depende de los casos de uso "A" y "B".

**Extensión (extend):** Es otra forma de interacción, una extensión representa una variación de un caso de uso a otro, es decir, una dependencia específica entre los casos de uso, a través de la cual un caso de uso puede extender a otro.

**3.2.3 Diagramas de secuencias.** Es aquel que muestra la forma en que los objetos interactúan entre sí al transcurrir el tiempo. Consta de objetos que se representan del modo usual: rectángulos con nombre (subrayado), mensajes representados por líneas continuas con una punta de flecha y el tiempo representado como una progresión vertical.

**Objetos:** Se ubican en la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen al diagrama. La línea que está debajo de cada objeto será una línea discontinua conocida como la línea de vida de un objeto.

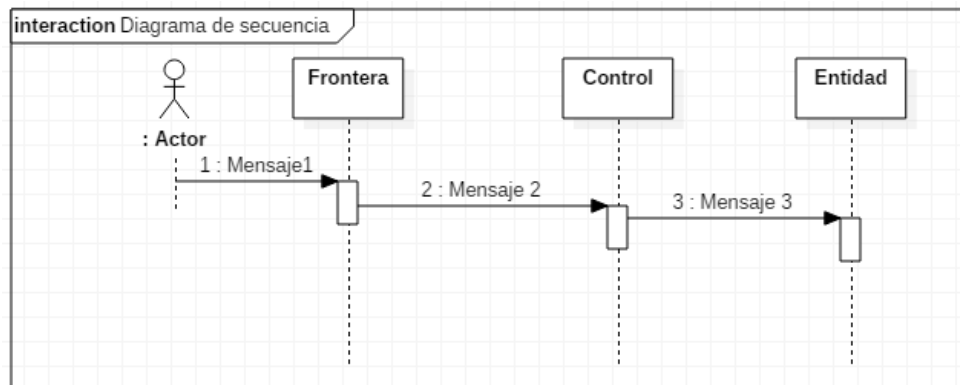
Con la línea de vida se encuentra un pequeño rectángulo conocido como activación, el cual representa la ejecución de una operación que realiza el objeto.

**Mensaje:** Un mensaje que va de un objeto a otro pasa la línea de vida de un objeto a otro. Un objeto puede enviarse un mensaje a sí mismo. Un mensaje puede ser simple, sincrónico o asíncrono.

**Tiempo:** El diagrama representa al tiempo en dirección vertical. Inicia en la parte superior y avanza hacia la parte inferior. Un mensaje que esté más cerca de la parte superior ocurrirá antes que uno que esté cerca de la parte inferior.

**GUI:** (Siglas en Ingles) La interfaz gráfica de usuario; es la interfaz de interacción del usuario y en la que más interactividades se presentan con otros objetos.

*Figura 5. Diagramas de secuencias.*



### 3.3 ESTÁNDARES DE PROGRAMACIÓN

**3.3.1 Modelo de datos.** Es un lenguaje utilizado para la descripción de una base de datos, por lo general permite describir estructuras de datos de la base de datos (el tipo de datos que incluye la base y la forma en que se relacionan), las restricciones de integridad (las condiciones que los datos deben cumplir para reflejar correctamente la realidad deseada) y las operaciones de manipulación de los datos (agregar, borrar, modificar).

**3.3.2 Nombres de las tablas.** Los nombres de los campos, así como de las tablas de la base de datos, se escriben en minúsculas, exceptuando los prefijos TP, TR y TB que indican si la tabla es principal, relacional o básica, respectivamente, y exceptuando también la primera letra de cada palabra que conforme su nombre; si es un nombre compuesto por dos o más palabras, los nombres tendrán en mayúscula la primera letra de cada palabra que la forma.

Como se mencionó, se han definido tres categorías para las diferentes tablas que conforman la base de datos. Dada la categoría de la tabla, se agrega un prefijo a su nombre que permita conocer la categoría a la que pertenece. Las categorías son:

- **Tabla básica:** Aquella cuyos registros son necesarios para el correcto funcionamiento de la base de datos. Estas tablas no experimentan muchos cambios en los datos. El prefijo para anteponer a los nombres de estas tablas es “TB\_”, es decir, la tabla que almacena las redes sociales que pertenecen a un grupo es llamada “TB\_GrupoRedSocial”, por ejemplo.
- **Tabla de Relación:** Surge de la relación muchos a muchos de una o dos tablas cualquiera. Los nombres de las tablas de relación deben ser siempre descriptivos para cada relación. El prefijo para anteponer a los nombres de estas tablas es “TR\_”, por ejemplo, la tabla “InscripcionesGrupo” que almacena las inscripciones creadas para los grupos de investigación “TR\_InscripcionesGrupos”.

- **Tabla Principal:** Aquella cuyo número de registros tiende a crecer en gran cantidad y que además no es posible clasificar como tabla básica o de relación. Un ejemplo de tabla principal es la tabla que almacena los usuarios del portal ESIWeb. El prefijo para anteponer a los nombres de estas tablas es “TP\_”, es decir la tabla “Usuarios”, es conocida como “TP\_ Usuarios”.

**3.3.3 Clases.** Los nombres de los campos, así como de las tablas de la base de datos, se escriben en minúsculas, exceptuando los prefijos TP, TR y TB que indican si la tabla es principal, relacional o básica, respectivamente, y exceptuando también la primera letra de cada palabra que conforme su nombre; si es un nombre compuesto por dos o más palabras, los nombres tendrán en mayúscula la primera letra de cada palabra que la forma.

**3.3.4 Páginas JSP.** Los nombres de las páginas JSP que componen los portales web comunidad académica son escritos de manera que la primera letra es una mayúscula seguido de letras minúsculas, en caso de que el nombre del JSP sea compuesto por dos o más palabras, entonces la primera de cada palabra debe ir en mayúscula, por ejemplo, SolicitarRetiroDirector.jsp, PeticionesAjaxProfesor.jsp.

**3.3.5 Organización de Directorios.** Los directorios del sitio están organizados de tal manera que los archivos que se almacenen en ellos correspondan a lo que describe el nombre del directorio.

## **4. DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO**

Para el desarrollo de este proyecto se siguió la metodología de prototipo evolutivo. Al iniciar el proyecto se elaboró un primer prototipo durante la fase de requerimientos, el cual fue mejorado con la inclusión de nuevos requerimientos surgidos en la fase de desarrollo, a medida que se generaba un prototipo, el mismo era sometido a pruebas de funcionamiento y se le realizaban los refinamientos pertinentes a partir del resultado de dichas pruebas.

### **4.1 PROTOTIPO ESPERADO**

Al iniciar el proyecto aún no se contaba con una concepción clara de lo sería el producto final, sin embargo, en el transcurso del desarrollo y evolución de los prototipos, las pruebas y análisis del sistema se pudo comprobar que se estaba acercando a los requerimientos iniciales, esto con el fin de enfocar el desarrollo a la solución de las necesidades de los usuarios.

**4.1.1 Requerimientos de objetivo** El objetivo específico inicial y los requisitos que surgieron se dieron gracias a la realización de prototipos y la realimentación con el cliente. Para cada objetivo se listaron los requerimientos detallados de este, los cuales se cumplieron para el prototipo final.

#### **4.1.1.1 Realizar actualización al Portal Web de Grupos.**

##### **Objetivo inicial:**

- Realizar la actualización al Portal Web de grupos que permita soportar nuevas funcionalidades y diferentes menús.

**Requisitos finales del objetivo:**

- Nueva página de entrada con posibilidad de publicar banner, texto explicativo y enlace como en el Portal de las Escuelas.
- El portal contara con funcionalidades tales como contador y estadísticas del dispositivo, país y ciudad, sistema operativo, y otros datos sobre los visitantes.

**4.1.1.2 Módulo de inscripciones al Portal Web de Grupos.****Objetivo inicial:**

- Realizar un soporte al Portal de Grupo para crear el módulo de inscripciones públicas, escuela y UIS para las diferentes necesidades del grupo como talleres, seminarios, congresos, eventos, nuevos miembros y otros.

**Requisitos finales del objetivo:**

- El portal contara con el módulo de inscripciones para los diferentes roles con la posibilidad de subir archivos.
- En el portal se abre la posibilidad de agregar inscritos como miembros del grupo para ingresar e interactuar. Esta labor la debe realizar solo los administradores de grupo.
- Este módulo debe permitir hacer seguimiento y la posibilidad de enviar correos entre los interesados inscritos y el administrador de la inscripción.

**4.1.1.3 Nuevo servicio que permita a los directores solicitar ante el Comité de Trabajos de Grado retirarse de un proyecto.****Objetivo inicial:**

- En el módulo de Trabajos de Grado realizar un nuevo servicio que permita a los Directores solicitar ante el Comité de Trabajos de Grado retirarse de un proyecto.

**Requisitos finales del objetivo:**

- Se debe hacer todo el soporte para que sea visto por el comité en sesión, comité en sesión individual, mostrarse en acta, mostrarse de manera correcta en la información del proyecto.
- Se debe permitir a los miembros del comité en la sesión virtual tomar una posición y hacer un comentario.
- Se debe permitir al director retractarse, es decir, retirar la solicitud hasta antes de la sesión del comité.

**4.1.1.4 Guardar fecha y comentario del evaluador de solicitud al emitir concepto y almacenarlo de tal forma que se muestren todos los comentarios emitidos anteriormente al comité y al evaluador.****Objetivo inicial:**

- Guardar fecha y comentario del evaluador de solicitud al emitir concepto y almacenarlo de tal forma que se muestren todos los comentarios emitidos anteriormente al comité y al evaluador.

**Requisitos finales del objetivo:**

- Guardar fecha y comentario del evaluador de solicitud al emitir concepto y almacenarlo de tal forma que se muestren todos los comentarios emitidos anteriormente al comité y al evaluador.

**4.1.1.5 Nuevo servicio de intercambio de mensajes mediante el correo electrónico que permita la comunicación entre autores y director al revisar el plan.**

**Objetivo inicial:**

Capturar y guardar el comentario realizado por el director a los autores al revisar el plan y enviarlo mediante correo electrónico.

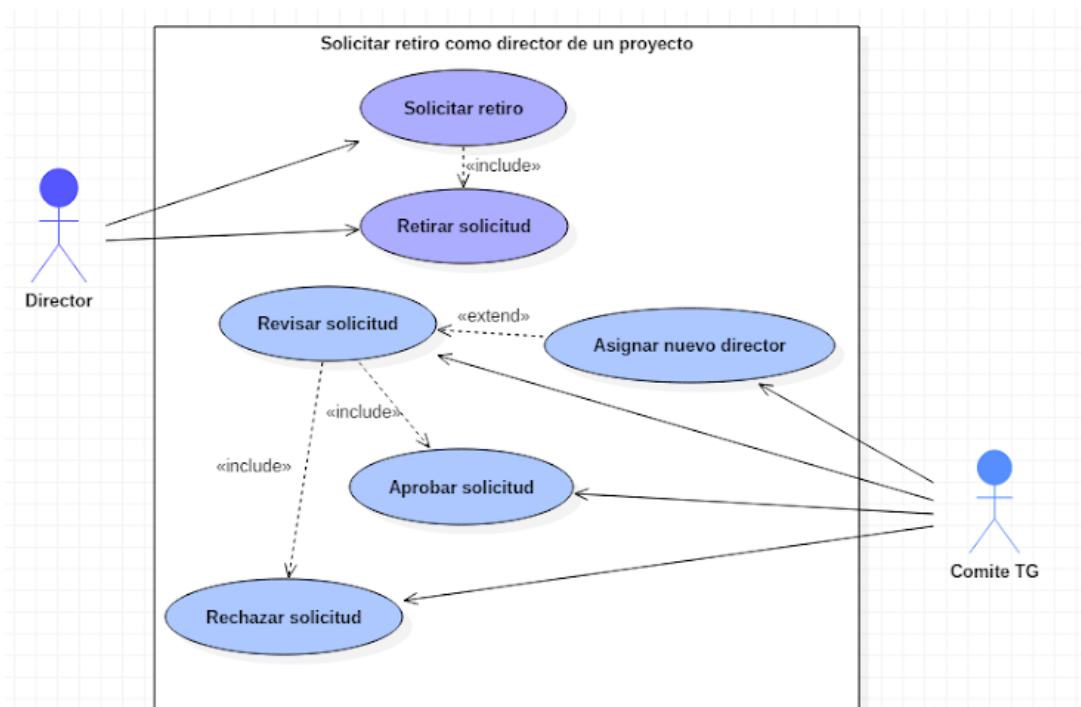
**Requisitos finales del objetivo:**

- El autor o los autores de un proyecto podrán comunicarse con su respectivo director de proyecto al inscribir, modificar y solicitar la aprobación de un plan, capturando hora y fecha, para mandarlo por correo electrónico.

## 4.2 DIAGRAMAS DE CASOS DE USO

### 4.2.1 Solicitar al comité de trabajos de grado retirarse de un proyecto como director

Figura 6. Diagrama de casos de uso: Solicitar retiro como director al comité de trabajos de grado.



## 4.2.2 Actualización al Portal Web de Grupos

Figura 7. Diagrama de casos de uso: Actualizar portal web de grupos.

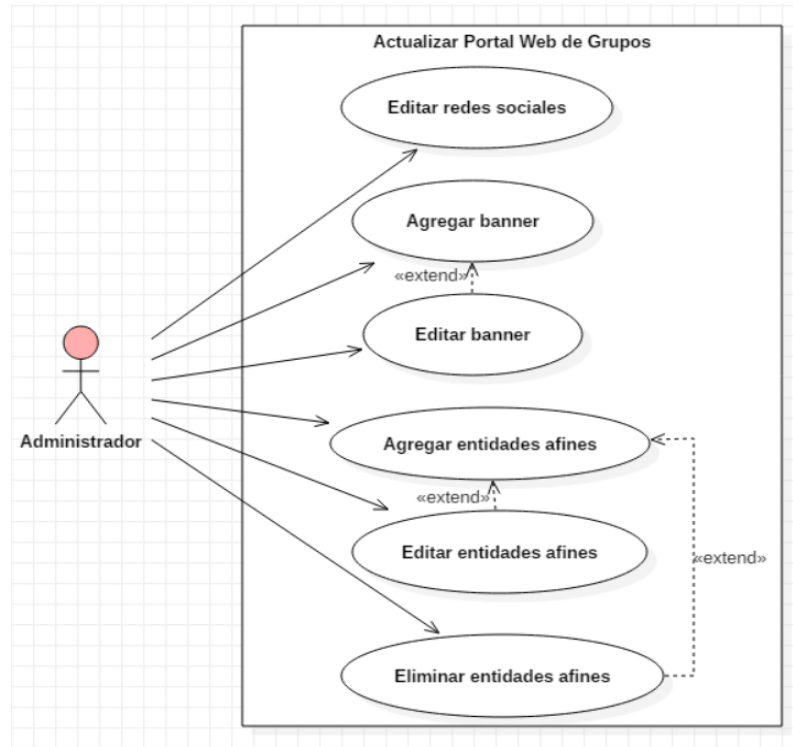
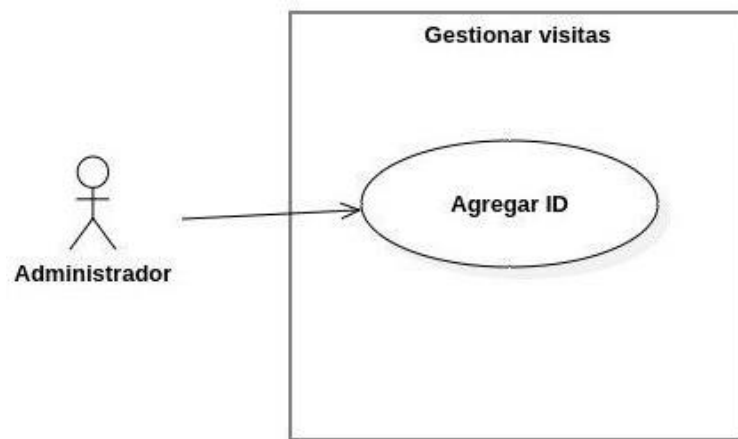


Figura 8. Gestionar visitas.



### 4.2.3 Inscripciones Portal Web Grupos

Figura 9. Diagrama de casos de uso: Administrar inscripciones Escuelas por un administrador e integrante del grupo.

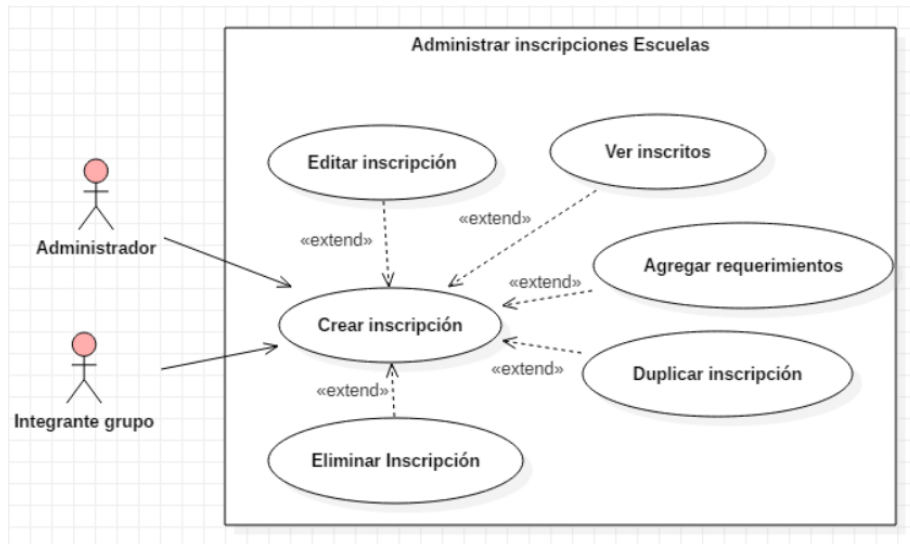


Figura 10. Diagrama de casos de uso: Inscripciones abiertas Escuelas por un integrante del grupo y un usuario externo.

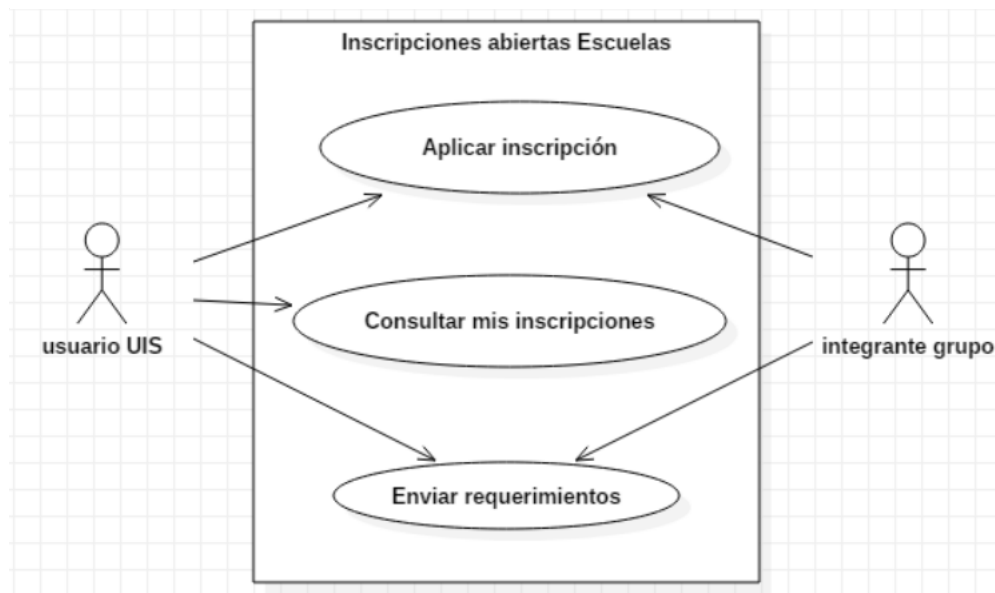


Figura 11. Diagrama de casos de uso: Administrar inscripciones UIS por un administrador e integrante del grupo.

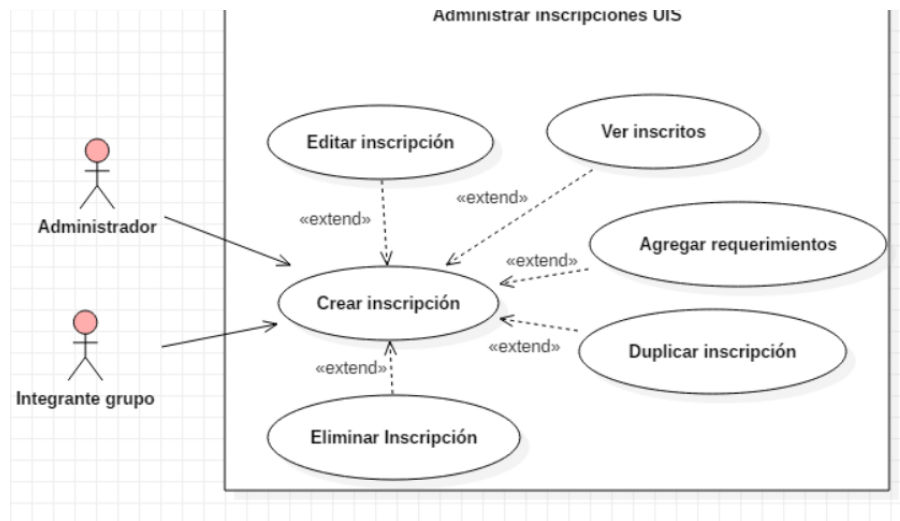


Figura 12. Diagrama de casos de uso: Ver inscritos escuela y ver inscritos comunidad UIS por un administrador e integrante del grupo.

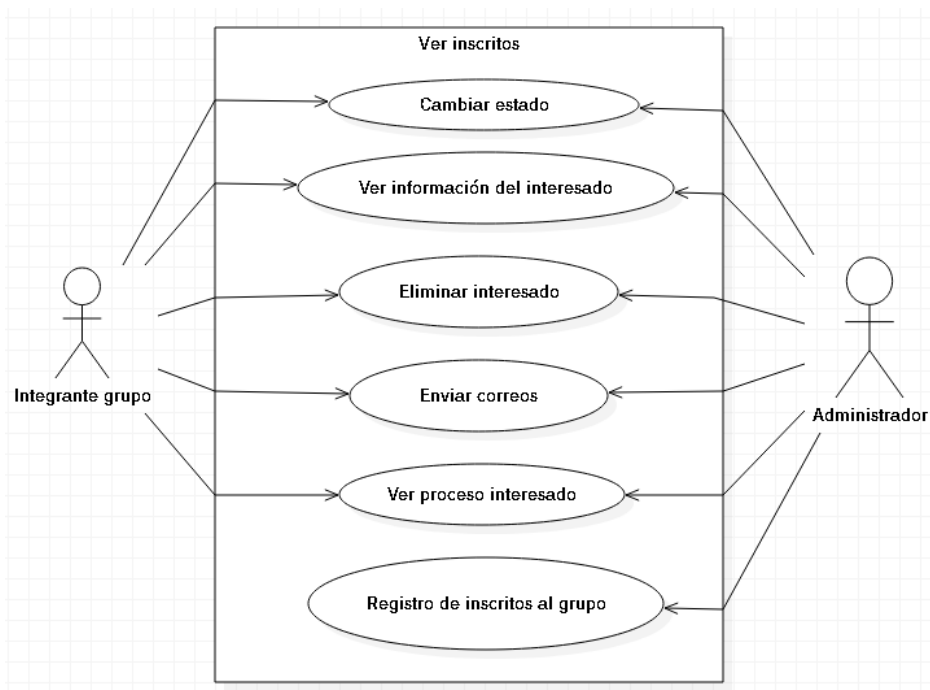
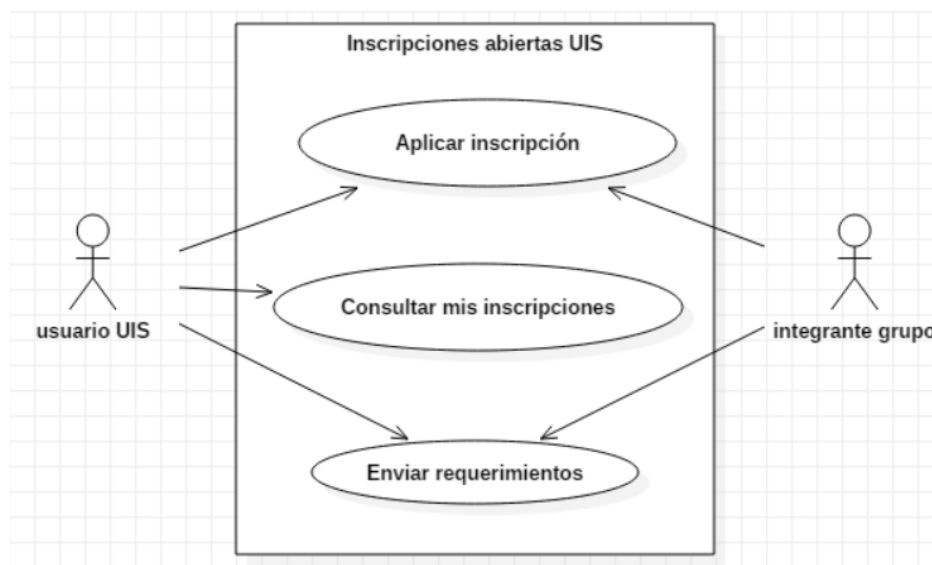


Figura 13. Diagrama de casos de uso: Inscripciones abiertas comunidad UIS por un integrante del grupo y un usuario UIS.



### 4.3 DOCUMENTACIÓN DE CASOS DE USO DEL SISTEMA

#### 4.3.1 Solicitar al comité de trabajos de grado retirarse de un proyecto como director

Tabla 1 Casos de uso: Solicitar retiro al comité de trabajo de grado como director.

TÍTULO	FUNCIONES PRIMARIAS
Solicitar retiro	El director de un proyecto de grado solicita el retiro como director de ese proyecto.
Retirar solicitud	El director se retracta de la solicitud.
Revisar solicitud	El comité TG revisa la solicitud realizada por el director.
Asignar nuevo director	Comité TG, asigna un nuevo director, ya sea un profesor o Director no asignado.

Aprobar solicitud	Comité TG aprueba la solicitud del director, asignando un nuevo director o director no asignado.
Rechazar solicitud	Comité TG rechaza la solicitud realizada por el director.

#### 4.3.2 Actualización al Portal Web de Grupos

Tabla 2 Casos de uso: Actualizar portal web de grupos.

TÍTULO	FUNCIONES PRIMARIAS
Editar redes sociales	El administrador puede incluir direcciones url de las redes sociales del grupo.
Agregar banner	El administrador puede agregar imágenes, texto principal, texto secundario, nombre del botón y url y puede hacer visible o no la imagen en el banner.
Editar banner	El administrador edita los textos, url, nombre del botón e imagen que desea.
Agregar entidades afines	El administrador agrega logotipo, url y descripción de la entidad que quiera mostrar.
Editar entidades afines	El administrador edita logotipo, url y descripción de la entidad a mostrar.
Eliminar entidades afines	El administrador elimina la entidad afín que no desee mostrar.

### 4.3.3 Actualización al Portal Web de Grupos

Tabla 3. Gestionar visitas

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Agregar identificador	El administrador puede agregar el identificador de la página del grupo, que permitirá acceder a las estadísticas de la misma desde Google Analytics.

### 4.3.4 Inscripciones Portal Web Grupos

Tabla 4 Casos de uso: Administrar inscripciones escuelas por un administrador e integrante del grupo.

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Crear inscripción	El administrador o integrante crea una inscripción de cualquier evento para los estudiantes de la escuela.
Editar inscripción	El administrador o integrante editan las opciones o preguntas de la inscripción.
Eliminar inscripción	El administrador o integrante elimina una inscripción que no se desea que se muestre en la escuela.
Ver inscritos	El administrador o integrante puede observar la cantidad de aspirantes inscritos al evento a realizar.
Agregar requerimientos	El administrador o integrante puede agregar ciertos requerimientos a una inscripción en específico que requiera la actividad, con los cuales debe cumplir el aspirante para seguir con el proceso.

Duplicar inscripción	El administrador o integrante puede duplicar una inscripción con las mismas características o especificaciones de la que está duplicando.
----------------------	---

Tabla 5 Casos de uso: Inscripciones abiertas escuelas por un integrante del grupo y usuario externo.

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Ver inscripciones inscrito	El integrante o usuario puede ver las inscripciones a las que se ha inscrito con anterioridad.
Ver inscripciones abiertas	El integrante o usuario pueden ver las inscripciones abiertas del grupo para la escuela a la que desean aplicar.
Digitar inscripción	El integrante o usuario digita los campos requeridos en la inscripción para la actividad que deseen participar.
Enviar correo	El integrante o usuario puede enviar un correo para comunicarse con el personal encargado.
Subir archivo	El integrante o usuario puede subir un archivo si lo requieren las especificaciones de la inscripción.

Tabla 6 Casos de uso: Administrar inscripciones comunidad UIS por un administrador e integrante del grupo.

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Crear inscripción	El administrador o integrante crea una inscripción de cualquier evento para los estudiantes de la comunidad UIS.

Editar inscripción	El administrador o integrante editan las opciones o preguntas de la inscripción.
Eliminar inscripción	El administrador o integrante elimina una inscripción que no se desea que se muestre a la comunidad UIS.
Ver inscritos	El administrador o integrante puede observar la cantidad de aspirantes inscritos al evento a realizar.
Agregar requerimientos	El administrador o integrante puede agregar ciertos requerimientos a una inscripción en específico que requiera la actividad, con los cuales debe cumplir el aspirante para seguir con el proceso.

Tabla 7 Caso de uso: Ver inscritos escuela y ver inscritos UIS por el administrador e integrante del grupo

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Cambiar estado	El administrador o integrante cambia el estado de algún interesado que haya aplicado a una inscripción.
Ver información del interesado	El administrador o integrante puede ver y actualizar la información de algún interesado que haya aplicado a la inscripción.
Eliminar interesado	El administrador o integrante elimina a un interesado que considere no cumpla los requisitos de la inscripción a la cual aplico.
Enviar correos	El administrador o integrante enviar correo a uno o varios interesados de la inscripción a la cual aplico.

Ver proceso interesado	El administrador o integrante puede verificar que los requerimientos de la inscripción a la cual aplico el interesado se hayan cumplido.
Registro de inscritos al grupo	El administrador registrar como usuarios del grupo a aquellos interesados cuyo estado sea 'Inscrito' en una inscripción.

Tabla 8 Casos de uso: Inscripciones abiertas comunidad UIS por un integrante del grupo y usuarios de la o las escuelas a las que valla dirigidas dicha inscripción.

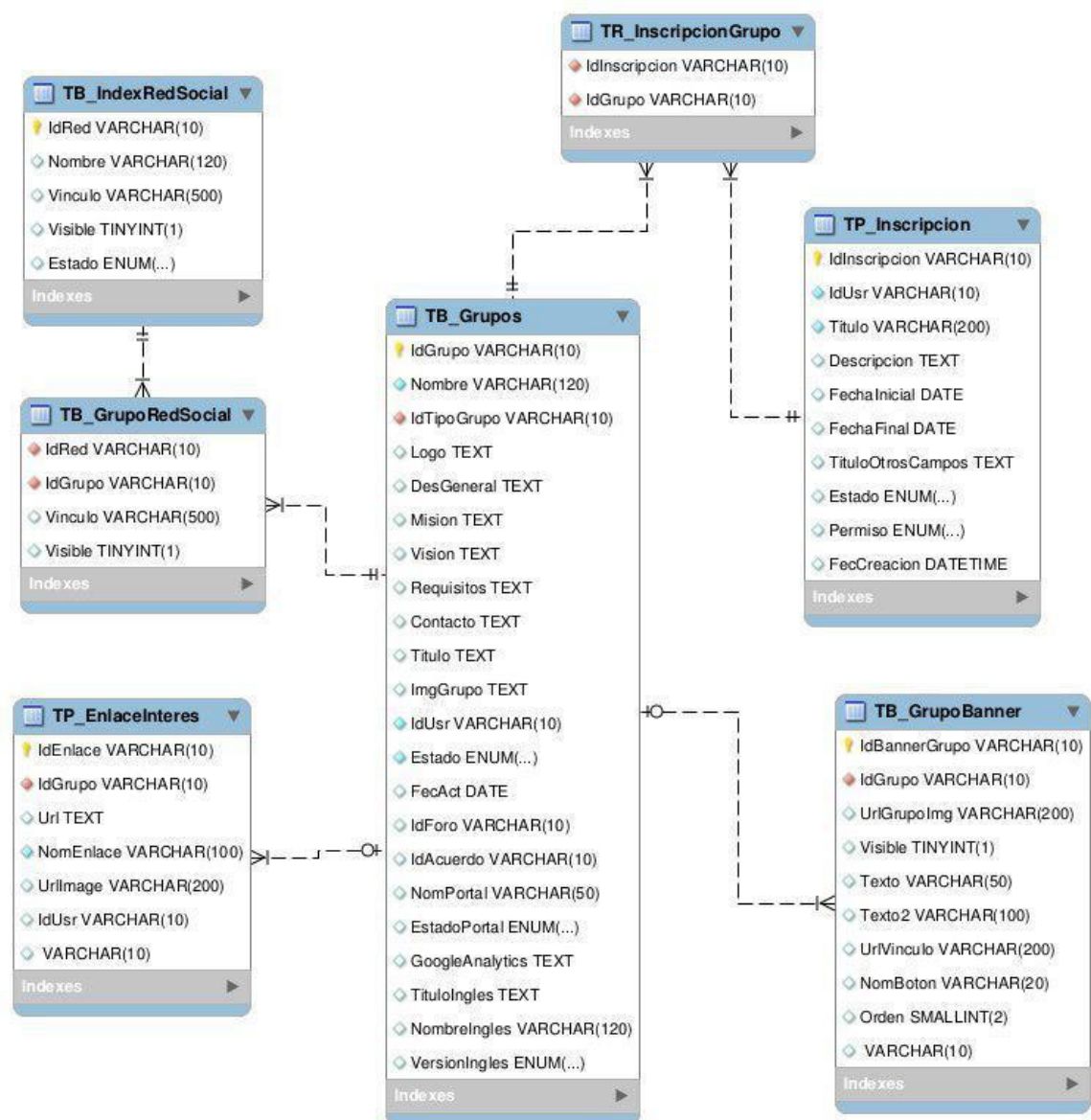
<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Ver inscripciones inscrito	El integrante o usuario puede ver las inscripciones a las que se ha inscripto con anterioridad.
Ver inscripciones abiertas	El integrante o usuario pueden ver las inscripciones abiertas del grupo para la comunidad UIS a la que desean aplicar.
Digitar inscripción	El integrante o usuario digita los campos requeridos en la inscripción para la actividad que deseen participar.
Enviar correo	El integrante o usuario puede enviar un correo para comunicarse con el personal encargado.
Subir archivo	El integrante o usuario puede subir un archivo si lo requieren las especificaciones de la inscripción.

## 4.4 DISEÑO Y ANÁLISIS

### 4.4.1 Diagrama Entidad/Relación de los servicios desarrollados

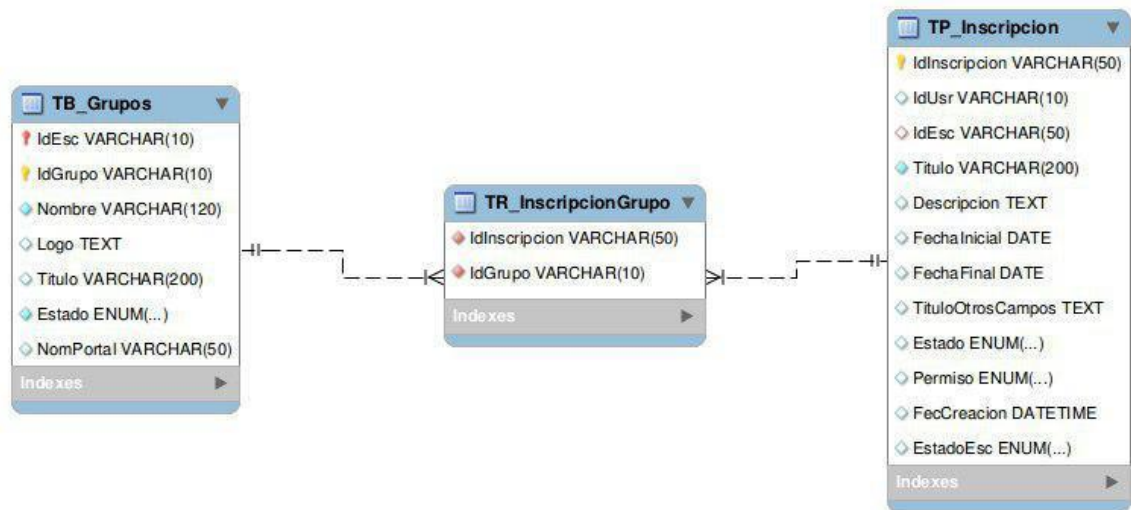
#### 4.4.1.1 Servicio Grupos Comunidad Escuela

Figura 14. Diagrama E/R: Portal de grupos e inscripciones grupos.



#### 4.4.1.2 Servicio Grupos Comunidad UIS

Figura 15. Diagrama E/R: Inscripciones comunidad UIS de grupos.



#### 4.4.1.3 Servicio Trabajos de Grado

Figura 16. Diagrama E/R: Solicitar retiro como director de proyecto.



#### 4.5 DESCRIPCIÓN DE LAS ENTIDADES.

Descripción de las entidades creadas y utilizadas en las bases de datos Diamante ubicada en el servidor de cada portal y Poseidón ubicada en el servidor del portal web de la Vicerrectoría Académica.

Tabla 9. Descripción de las entidades.

ENTIDAD	DESCRIPCION
<b>TB_GrupoRedSocial</b>	Contiene la información de las redes sociales del grupo.
<b>TB_GrupoBanner</b>	Contiene la información de las imágenes subidas para mostrar en el banner.
<b>TR_InscripcionGrupo</b>	Relaciona las inscripciones creadas por los grupos dirigida a la escuela que pertenece.
<b>TR_InscripcionGrupo</b>	Relaciona las inscripciones creadas por los grupos que están dirigidas a la comunidad UIS.
<b>TR_Directores</b>	Relaciona los directores con los proyectos de grado.

#### 4.6 MODELO DE PROCESOS DEL SISTEMA.

Para una mejor interpretación de los modelos de los procesos del sistema se realizaron los diagramas de secuencia necesarios para cada caso de uso en los que se explica con detalle los pasos para el funcionamiento de cada uno de los servicios (Ver Anexo A).

## 4.7 IMPLEMENTACIÓN, IMPLANTACIÓN Y PRUEBAS GENERALES

Para la implementación de los servicios se utilizaron las siguientes herramientas:

- Lenguaje de marcado para la elaboración de páginas web, HTML.
- Lenguaje de programación orientado a la web, JSP.
- Lenguaje de programación interpretado orientado a objetos, JavaScript.
- Hojas de estilo en cascada, CSS.
- Lenguaje Java.
- NetBeans, IDE para desarrollar las clases de Java y los archivos JSP.
- Servidor Jakarta Tomcat.
- Manejador de Base de datos, MySQL 5.0.

Haciendo uso de las anteriores herramientas, y con la asesoría y seguimiento del director de proyecto se diseñaron los nuevos servicios y se estructuraron los datos para el primer prototipo. También se tomaron en cuenta sugerencias recibidas por miembros del Grupo Calumet, para realizar un posterior refinamiento a las interfaces.

Para el desarrollo se trabajó con la base de datos "Diamante" ubicada en todos los servidores de los portales de las escuelas, y con la base de datos centralizada "Poseidón" ubicada en el servidor del portal web de la Vicerrectoría Académica. En las cuales se crearon las tablas necesarias para el funcionamiento de los servicios. Se llevaron a cabo pruebas para cada subsistema propuesto, verificando que el resultado correspondiera con lo esperado, de esta forma, se evidenció el correcto funcionamiento en la captura de datos, selección de ítems y almacenamiento de información.

La implantación de los servicios se realizó en primer lugar en el portal web de pruebas del Grupo Calumet, donde se realizan las pruebas a todos los componentes. Finalmente se implementan en todos los portales web a los cuales el Grupo Calumet presta soporte.

## **4.8 MANTENIMIENTO Y ADMINISTRACIÓN**

**4.8.1 Actividades de Mantenimiento.** Dentro de las funciones que se realizan en la administración y mantenimiento de los servicios de los portales web se encuentra la tarea de corregir errores, las cuales se denominan incidencias, que se corrigen durante la primera fase como integrantes del Grupo Calumet.

**4.8.2 Actividades de Soporte a Usuarios.** Las escuelas cuentan en sus portales con un servicio de consultas y sugerencias, a través del cual se responden preguntas y se resuelven problemas de los usuarios. Diariamente se reciben consultas de estudiantes y profesores, que necesitan asesoría en el funcionamiento de servicios o soporte en el manejo de su usuario y contraseña. También se resuelven problemas de manera presencial, donde el usuario acude directamente a las oficinas del Grupo Calumet y es atendido directamente por alguno de sus miembros, que le prestan la asesoría necesaria. Algunas de las labores realizadas comúnmente son:

- Restablecimiento de contraseña a usuarios.
- Dar respuesta y solución a las inquietudes y problemas que tengan los usuarios de los portales respecto al uso de los servicios.
- Facilitar orientación a los estudiantes nuevos de las escuelas a las cuales presta sus servicios el grupo Calumet, en cuanto al registro en el portal y el uso de sus principales servicios.

**4.8.3 Actividades de Administración.** Dentro de las actividades realizadas por los miembros del Grupo Calumet, se encuentra la tarea de administrar los servidores de los portales de las escuelas, cada semestre se asignan nuevos administradores, los cuales cuentan con un usuario dentro de uno de los portales a los cuales presta soporte el grupo. Como administrador las tareas que se realizan comúnmente son:

- Realizar periódicamente copias de las bases de datos Diamante y División.
- Dar aval a las solicitudes de publicación de los usuarios en la cartelera para que puedan ser accedidas por la comunidad.
- Atender las sugerencias hechas por los usuarios del sistema a través del servicio de consultas y sugerencias.
- Actualizar periódicamente las bases de datos con respecto a la información que ofrece la División de Servicios de Información de la Universidad Industrial de Santander, para que el portal cuente con información actualizada.

## **5 PRUEBAS DEL SISTEMA**

Para garantizar el correcto desarrollo de los nuevos servicios creados y a los que se le hizo reingeniería, se realizaron las siguientes pruebas:

### **5.1 PRUEBAS DE VERIFICACIÓN**

Esta prueba es una de las más utilizadas en desarrollo de software mediante esta se aplican diferentes técnicas para detectar errores en el sistema antes de ser utilizado.

Se efectúa ejecutando paso a paso el proceso del servicio de manera que se explora cada funcionalidad que tiene el módulo desarrollado, realizando verificaciones de validación, los campos que son obligatorios no pueden quedar vacíos, por ejemplo.

Las siguientes tablas describen las pruebas de cada caso de uso de los servicios que fueron desarrollados.

### 5.1.1 Pruebas por componente.

Esta prueba se realizó para cada caso de uso de cada servicio desarrollado y descrito anteriormente:

#### 5.1.1.1 Solicitar al comité de trabajos de grado retirarse de un proyecto como director

Tabla 10. Pruebas realizadas: Solicitar retiro como director de un proyecto.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Solicitar retiro	Ingresar por trámites de proyectos en la pestaña solicitar retiro como director, selecciona el proyecto de grado al cual pertenezca y solicitar el retiro al comité TG.	✓
Retirar solicitud	Seleccionar el proyecto del cual solicito retirarse y retractar la solicitud.	✓
Revisar solicitud	Ingresar por sesión comité TG, revisar la solicitud enviada por el director del proyecto.	✓
Asignar nuevo director	Al revisar la solicitud el comité puede asignar un profesor para dirigir el proyecto o en caso de no tenerlo selecciona director no asignado.	✓
Aprobar solicitud	Con un profesor o con Director No Asignado el comité TG aprueba la solicitud.	✓
Rechazar solicitud	Rechaza la solicitud realizada por el director del proyecto.	✓

### 5.1.1.2 Actualización al Portal Web de Grupos

Tabla 11. Pruebas realizadas: Actualizar portal web grupos.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Editar redes sociales	Editar la url de las redes sociales del grupo.	✓
Agregar banner	Agregar imágenes, texto principal, texto secundario, url y hacer visible o no para que se vea en la página de inicio del grupo.	✓
Editar banner	Editar imagen, textos, url para que sea visible en el portal del grupo.	✓
Agregar entidades afines	Agregar imagen, nombre y url de la entidad que desea hacer visible en el portal del grupo.	✓
Editar entidades afines	Editar la imagen, nombre y url de la entidad que dese mostrar.	✓
Eliminar entidad interesada	Eliminar imagen, nombre y url de la entidad que no se desea mostrar.	✓

### 5.1.1.3 Inscripciones Portal Web Grupos.

Tabla 12. Pruebas realizadas: Administrar inscripciones escuelas.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Crear inscripción	Listar los requerimientos que desea incluir en la inscripción y crearla para ser visible en la escuela del grupo.	✓
Editar inscripción	Actualizar la información de la inscripción ya creada, y verificar los cambios realizados.	✓
Eliminar inscripción	Verificar que se elimine la inscripción de la escuela a la que pertenece el grupo.	✓
Ver inscritos	Visualizar y verificar la cantidad de aspirantes inscritos.	✓

Agregar requerimiento	Digitar los requerimientos que se necesita en la inscripción y verificar que se visualicen a los aspirantes.	✓
Duplicar inscripción	Verificar que al duplicar la inscripción se visualice en la escuela a la que pertenece el grupo con los requerimientos correspondientes.	✓

Tabla 13. Pruebas realizadas: Inscripciones abiertas escuelas.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Ver inscripciones inscritos	Se listan las inscripciones a la cual ya se ha inscrito el aspirante.	✓
Ver inscripciones abiertas	Se listan las inscripciones que se han creado por parte del grupo para la comunidad de la escuela	✓
Digitar inscripción	Se digitan los campos requeridos de la inscripción y se avanza en el proceso.	✓
Enviar correo	Se envía correo para comunicarse con el personal encargado	✓
Subir archivos	Se suben archivos para los requerimientos solicitados por la inscripción.	✓

Tabla 14 Pruebas realizadas: Administrar inscripciones comunidad UIS

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Crear inscripción	Listar los requerimientos que desea incluir en la inscripción y crearla para ser visible en la escuela del grupo.	✓
Editar inscripción	Actualizar la información de la inscripción ya creada, y verificar los cambios realizados.	✓
Eliminar inscripción	Verificar que se elimine la inscripción de la escuela a la que pertenece el grupo.	✓

Ver inscritos	Visualizar y verificar la cantidad de aspirantes inscritos.	✓
Agregar requerimiento	Digitar los requerimientos que se necesita en la inscripción y verificar que se visualicen a los aspirantes.	✓
Duplicar inscripción	Verificar que al duplicar la inscripción se visualice en la escuela a la que pertenece el grupo con los requerimientos correspondientes.	✓

Tabla 15. Pruebas realizadas: Ver inscritos comunidad UIS y Escuela en grupos.

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>	
Cambiar estado	Crear una inscripción a la cual aplican varias personas y verificar que se listen en la pestaña correspondiente al estado en el cual se encuentren los interesados cuando el creador de la inscripción o el administrador lo modifique.	✓
Ver información del interesado	Acceder a la información del interesado verificando que se muestre correctamente y pueda ser actualizada por el creador de la inscripción o el administrador.	✓
Eliminar interesado	Eliminar a un interesado en cual sea de los estados que se encuentre.	✓
Enviar correos	Seleccionar uno o varios interesados a los que se desee enviar un correo y se agreguen a la caja de redacción de correos y ejecutar la acción de envío de correo a los destinatarios seleccionados.	✓
Ver proceso interesado	Acceder al formulario de ver proceso del interesado y verificar que este haya cargado los archivos correspondientes a	✓

	los requerimientos de la inscripción por parte del interesado, y poder validar en la caja de requerimientos que se cumplan.	
Registro de inscritos al grupo	El administrador del grupo debe poder ver el listado de inscritos en la pestaña de Registro de inscritos distinguiendo entre quienes pertenecen ya al grupo y los que no. Él debe poder seleccionar aquellos inscritos que no se encuentran registrados en el grupo y desee pertenezca a este con un determinado tipo de usuario para cada uno.	✓

Tabla 16. Pruebas realizadas: Inscripciones abiertas comunidad UIS

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Ver inscripciones inscritos	Se listan las inscripciones a la cual ya se ha inscrito el aspirante.	✓
Ver inscripciones abiertas	Se listan las inscripciones que se han creado por parte del grupo para la comunidad de la escuela	✓
Digitar inscripción	Se digitan los campos requeridos de la inscripción y se avanza en el proceso.	✓
Enviar correo	Se envía correo para comunicarse con el personal encargado	✓
Subir archivos	Se suben archivos para los requerimientos solicitados por la inscripción.	✓

## 6. CONCLUSIONES

- Se efectuó una nueva actualización en el portal Web de grupos, que consiste en la inserción de un banner apoyado en textos explicativos, acceso a páginas de interés y posibles redes sociales, visualización de eventos y noticias optimizando el espacio entre estos, logrando captar la atención de cada integrante del grupo. Por otro lado, mejora el diseño y la usabilidad del grupo, puesto que se utiliza como una estrategia de divulgación, resaltando de forma clara, concreta y sencilla lo importante.
- Cumpliendo con la evaluación y verificación del tráfico que llega al portal de grupos, el administrador del grupo debe solicitar ante el grupo CALUMET la inclusión de la herramienta de análisis web de la empresa de google llamada Google Analytics para analizar la audiencia, adquisición, comportamiento del portal web de grupos, para visualizar el número de visitantes que tiene el portal web clasificando por ubicación geográfica y medio de acceso.
- El servicio de inscripciones del portal de grupos para la comunidad de la escuela y la comunidad UIS, le permite a un integrante del grupo y/o administrador efectuar inscripciones de usuarios a las diferentes actividades (cursos, eventos, congresos, etc) que son promovidas por el grupo o, proporcionando del inscrito así como el envió de correos y la posibilidad de intercambiar documentos entre administrador e inscritos.
- En relación a lo antes expuesto, después de aceptar al inscrito en el portal de grupos, el administrador cumpliendo su función de ayudar con la organización, debe evaluar, comprobar y confirmar la posible agregación de un inscrito que no esté registrada en esa escuela para su registro en el grupo garantizando su posterior interacción en el mismo.

- Con la implementación del almacenamiento de los comentarios que los evaluadores emiten se garantiza el historial de estos, para su verificación por parte de autores, evaluadores o comité de trabajos de grado.
- La implementación de una sección donde el director pueda generar comentarios al respecto sobre el plan de trabajo de grado subido por el autor, permite una realimentación entre autor y director acerca del plan facilitando la comunicación entre ellos.
- El servicio de retirarse como director de un proyecto, en el módulo de trabajos de grados, facilita al director manifestar su decisión de retirarse del proyecto de una forma más rápida y sencilla, para que el comité pueda aceptar o rechazar la misma.
- La metodología de trabajo del grupo Calumet permite que los estudiantes brinden apoyo de manera continua al proyecto COMA a través de las capacitaciones a estudiantes nuevos y el manejo de estándares. Haciendo continuo uso del trabajo grupal ocasionando así una mejora continua y soporte sobre el proyecto sin perder la esencia del mismo.

## **7. RECOMENDACIONES**

Socializar con los integrantes de los grupos de investigación, acerca de la nueva versión de la página inicial del portal de grupos y brindar capacitación sobre el manejo de los servicios en la página inicial, para que así puedan hacer uso de manera óptima de este servicio.

Capacitar a los usuarios de los grupos sobre el nuevo servicio de inscripciones para que puedan hacer buen uso de este y logren ahorrar tiempo, agilizando tareas relacionadas con los mismos.

Socializar a los miembros del comité de trabajos de grado y profesores sobre el nuevo servicio de retirarse como director de un proyecto de grado, para que puedan hacer buen uso de este.

Realizar una actualización que permita la visualización del portal de grupos en dispositivos móviles.

## BIBLIOGRAFÍA

JAVASCRIPT-YA. Tutoriales del lenguaje de programación JavaScript. [En línea] Disponible en: <http://www.tutorialesprogramacionya.com/javascriptya/index.php?inicio=0>>.

JOHNSON, James. Bases de datos: modelos lenguajes, diseño. 1ª ed. México: Oxford University Press, 2000. 1015. ISBN: 970-613-461-1.

MANUALES. Manuales básicos de JSP. [En línea] Disponible en: <http://www.desarrolloweb.com/manuales/73/> >.

PIATTINI VELTHUIS, Mario, et al. Análisis y diseño detallado de Aplicaciones Informáticas de Gestión. Madrid: RA-MA S.A. Editorial y Publicaciones, 2007, 480. ISBN-10: 8478977767

PROGRAMACIÓN. Curso para aprender java. [En línea] Disponible en: [http://www.aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=68&Itemid=188](http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=68&Itemid=188) >.

PROGRAMACIÓN FÁCIL. Manual de programación para Java Jsp (Java Server Pages). [En línea] Disponible en: [http://www.programacionfacil.com/programacion:manual\\_java\\_jsp](http://www.programacionfacil.com/programacion:manual_java_jsp)>.

SCHMULLER, Joseph. Aprendiendo UML en 24 horas. México: Alhambra Mexicana S.A, 2000, 387.

SQL. Documentación con ejemplos de SQL. [En línea] Disponible en:  
<http://www.1keydata.com/es/sql/> >

STALLINGS, William. Sistemas Operativos: Aspectos Internos y Principios de Diseño. Madrid: Pearson Prentice Hall, 2005, 208. ISBN-13: 978-0-13-230998-1

STUMPF, Robert, TEAGUE, Lavette. Object-Oriented Systems Analysis and Design with UML. New Jersey: Pearson Prentice Hall, 2004, 428. . ISBN 0131434063

WEITZENFELD, Alfredo. Ingeniería de Software Orientada a Objetos con UML, JAVA e Internet. México: Thomson International, 2003, 678.

WIKISPACES. Información acerca de la Arquitectura Cliente Servidor. [En línea] Disponible en: <http://g701giadar.wikispaces.com/Arquitctura+Cliente+Servidor>>.

W3SCHOOLS. Tutoriales de cómo programar en HTML. [En línea] Disponible en:  
<http://www.w3schools.com/html/>>.

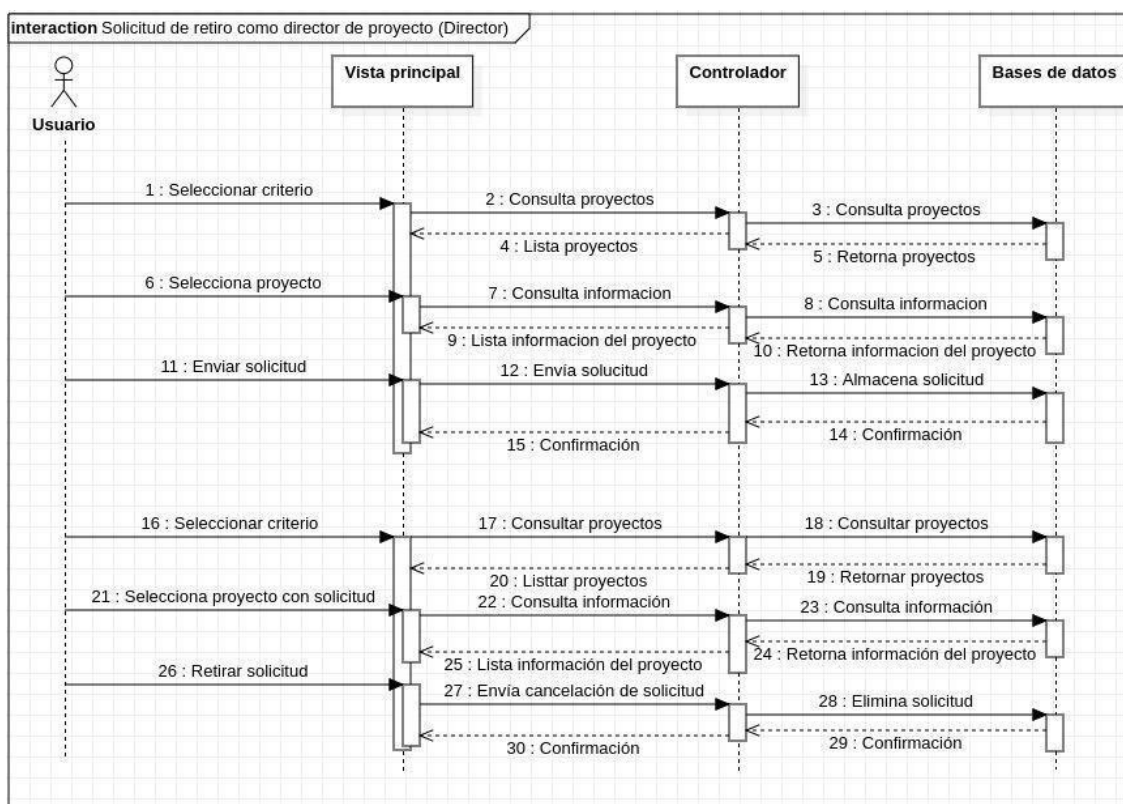
YOUBLISHER. Tutoriales básicos sobre UML. [En línea] Disponible en:  
<http://www.tutorialspoint.com/uml/> >

## ANEXOS

### Anexo A. Modelo de Procesos del Sistema

#### Servicio solicitar al Comité TG retirarse como Director de proyecto

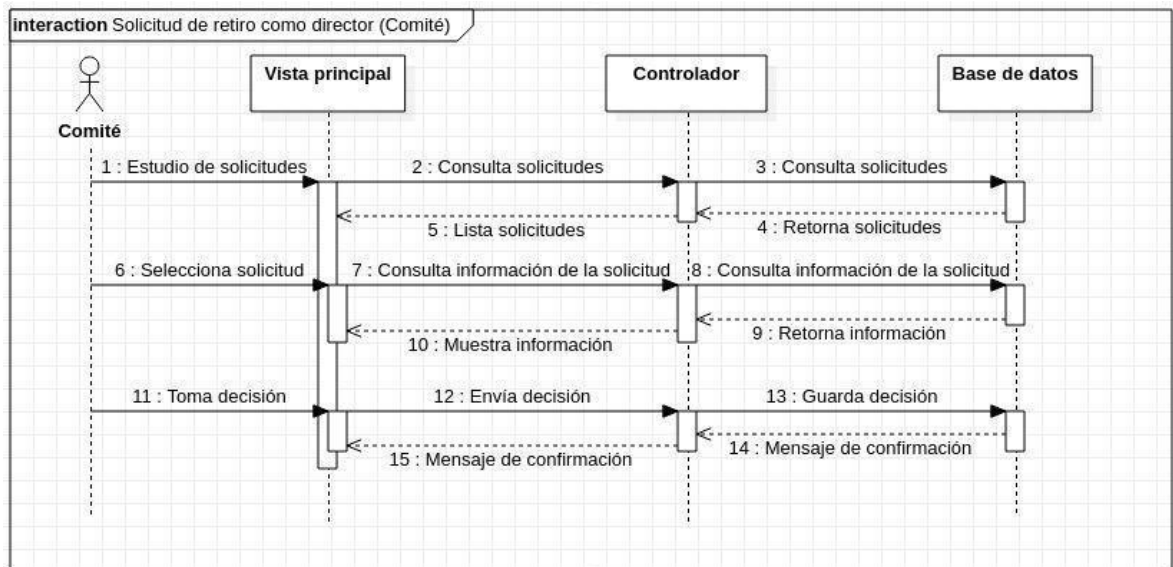
Figura 17. Diagrama de secuencia: Solicitar retiro como Director.



1. El Director selecciona el criterio de la solicitud en la Vista Principal.
2. La Vista Principal consulta al Controlador los proyectos del director.
3. El controlador valida y envía a la Base de Datos los proyectos del director.
4. La Base de Datos retorna todos los proyectos de director.
5. El controlador devuelve la lista de los proyectos a la Vista principal.
6. El usuario selecciona un proyecto y consulta la información del proyecto al

- controlador que manda una consulta a la Base de datos.
7. La Base de Datos retorna la información del proyecto al controlador.
  8. El Controlador lista la información del proyecto a la Vista Principal.
  9. El Director envía una solicitud de retiro del proyecto seleccionado.
  10. El controlador verifica los datos y manda la solicitud a la Base de Datos.
  11. La Base de Datos almacena la solicitud y envía un mensaje de confirmación al controlador.
  12. El controlador le envía un mensaje de confirmación a la Vista Principal.

Figura 18. Diagrama de secuencia: Retiro como Director sesión del Comité.

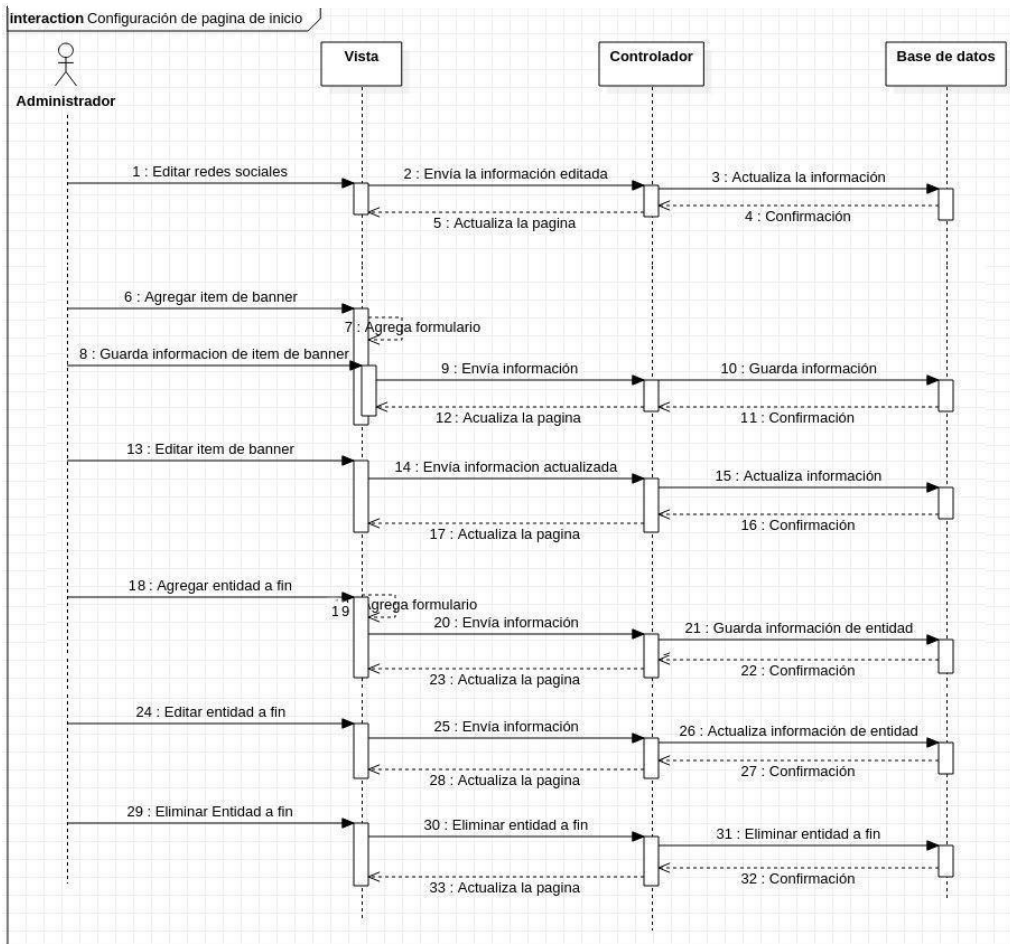


1. El Comité selecciona en el menú estudio de solicitudes en la Vista Principal.
2. La Vista Principal consulta las solicitudes al controlador.
3. El Controlador consulta las solicitudes a la Base de Datos.
4. La Base de Datos retorna las solicitudes al controlador.
5. El controlador devuelve una lista de las solicitudes a la vista principal.
6. El Comité selecciona una solicitud a la vista Principal.
7. La Vista principal consulta toda la información de dicha solicitud al controlador.

8. El controlador consulta la información de la solicitud a la Base de Datos.
9. La Base de Datos retorna la información de la solicitud al controlador.
10. El controlador recibe la información de la base de Datos y muestra a la Vista Principal la información solicitada.
11. El Comité toma una decisión con respecto a la solicitud del director en la Vista principal.
12. La vista principal valida los datos y envía la decisión al controlador.
13. La base de datos guarda la decisión de la solicitud.
14. La base de datos retorna un mensaje de confirmación al controlador.
15. El controlador envía la confirmación a la Vista Principal.

## Servicio Página de inicial del Portal de Grupos

Figura 19. Diagrama de secuencia: Configuración página de inicio.

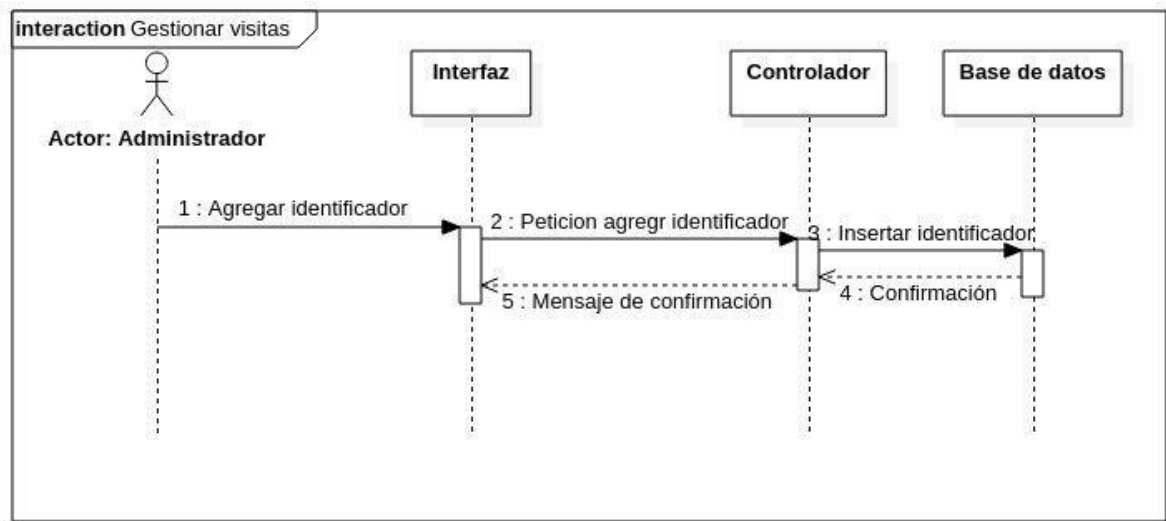


1. El Administrador edita las redes sociales asociadas al grupo en la Vista Principal.
2. La vista principal envía la información editada al controlador.
3. El controlador valida la información y la envía para ser almacenada en Base de datos.
4. La Base de Datos envía una confirmación de la información almacenada al Controlador.
5. El controlador actualiza la información en la vista principal.

6. El Administrador agrega una imagen para el banner en la Vista Principal.
7. La vista principal despliega un formulario para añadir la información al Administrador.
8. El Administrador guarda la información de la imagen en la vista principal.
9. La vista principal envía la información al controlador.
10. El controlador valida los datos y envía la información a Base de Datos.
11. La base de datos almacena la información y envía un mensaje de confirmación al controlador.
12. El controlador actualiza la información en la Vista Principal.
13. El Administrador edita la imagen e información del banner en la vista principal.
14. La vista principal envía la información actualizada al controlador.
15. El controlador valida los datos y envía la información a Base de datos.
16. La base de datos almacena la información y envía un mensaje de confirmación al controlador.
17. El controlador actualiza la información en la vista principal.
18. El administrador da clic en agregar una entidad a fin en la vista principal.
19. La vista principal le despliega un formulario, para agregar descripción y la dirección URL de la entidad.
20. El administrador digita los datos y la vista principal envía la información al controlador.
21. El controlador valida la información y la envía a la Base de datos.
22. La Base de datos almacena la información y envía una confirmación al controlador.
23. El controlador actualiza la Vista Principal.
24. El administrador edita la imagen y la información de la entidad en la vista principal.
25. La Vista principal envía la información al controlador.
26. EL controlador valida los datos y manda la información a la Base de Datos.
27. La base de datos almacena la información y envía una confirmación al controlador.
28. EL controlador actualiza la información en la vista principal.

- 29.El administrador elimina la imagen y la información de la entidad en la vista principal.
- 30.La Vista principal envía la información al controlador.
- 31.EL controlador valida los datos y manda la información a la Base de Datos.
- 32.La base de datos almacena la información y envía una confirmación al controlador.
- 33.EL controlador actualiza la información en la vista principal.

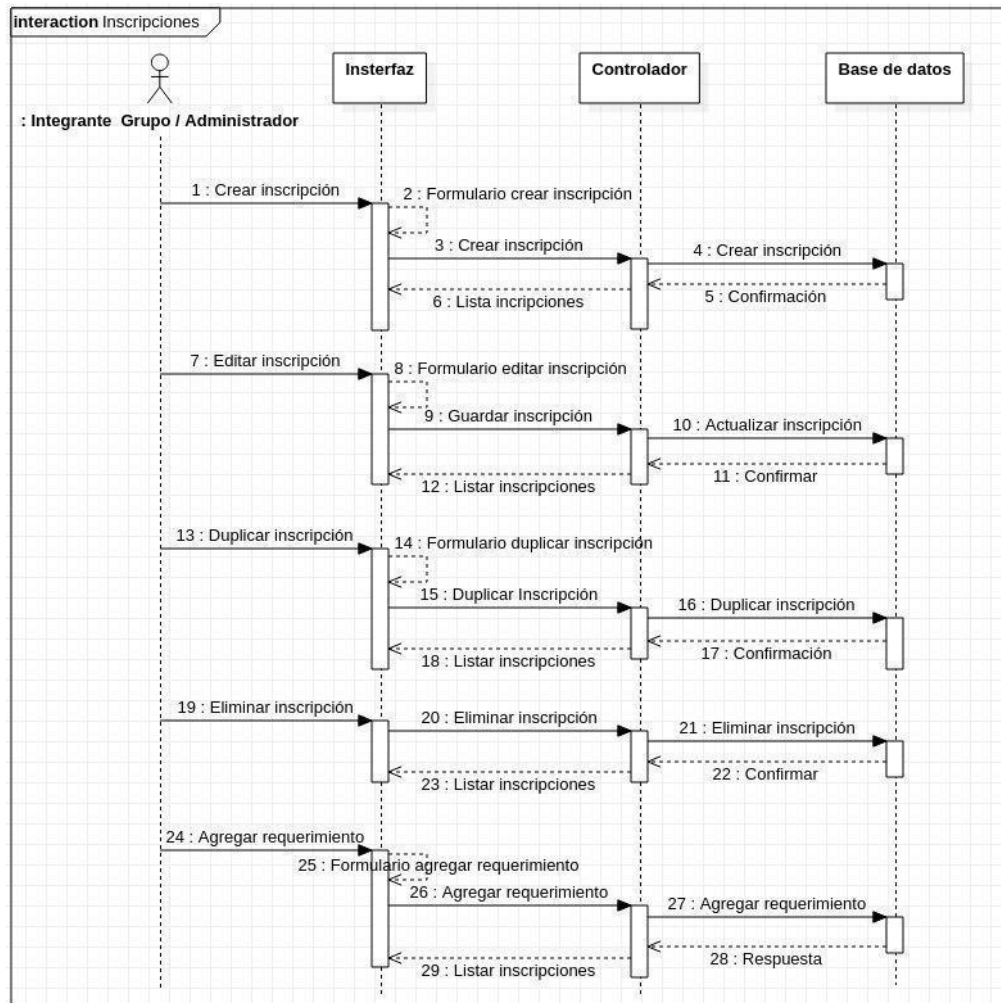
Figura 20. Diagrama de secuencia: Gestionar visitas.



1. El administrador del grupo agrega el identificador que se obtuvo previamente de la página de Google Analytics para hacerle seguimiento al portal del grupo.
2. Se envía la petición de agregar identificador al controlador.
3. El controlador verifica el código y lo envía a base de datos.
4. Se realiza la inserción en base de datos y regresa un mensaje de confirmación.
5. El controlador envía un mensaje de confirmación que se muestra al usuario a través de la interfaz.

## Servicio Inscripciones Portal Grupos

Figura 21. Diagrama de secuencia: Inscripciones portal de grupos escuela y UIS

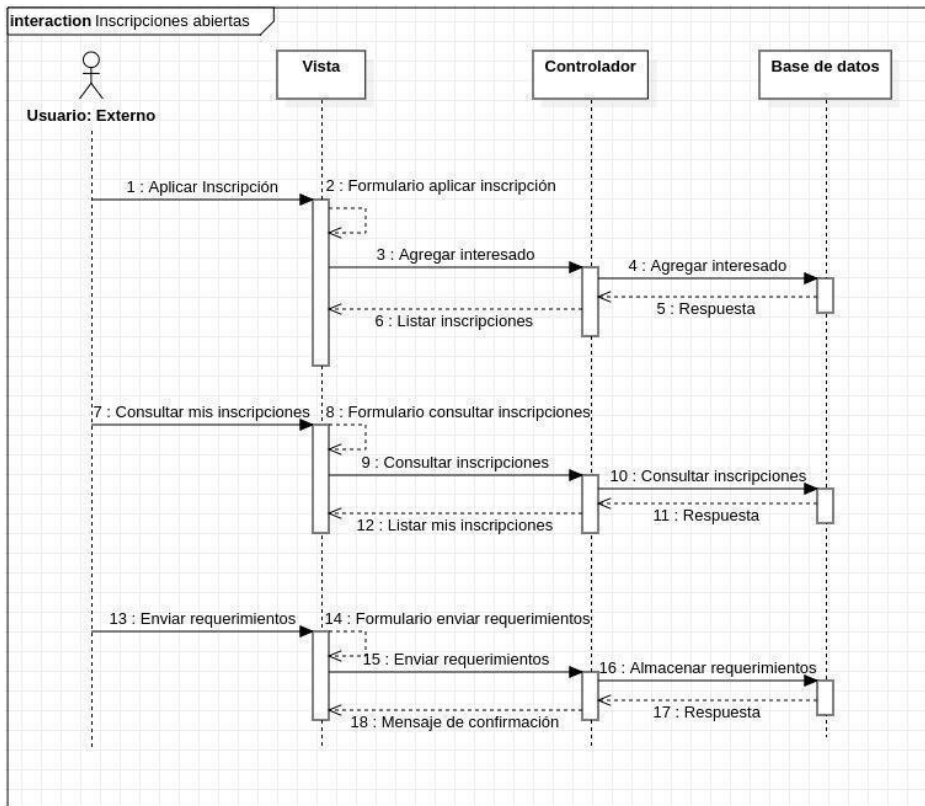


1. El administrador o integrante del grupo crea una inscripción.
2. La vista principal despliega un formulario de las opciones a elegir para la inscripción en la vista principal
3. La vista principal manda los datos de la inscripción al controlador para ser verificados.
4. El controlador verifica los datos que son correctos y los envía a la base de datos para ser almacenados.

5. La base de datos almacena la información y envía una confirmación al controlador.
6. El controlador lista todas las inscripciones creadas en la Vista principal.
7. El administrador o integrante de grupo puede editar las opciones o datos de la inscripción.
8. La vista principal le despliega el formulario con las opciones elegidas al momento de crear la inscripción para su edición por parte del administrador o integrante de grupo que guarda los datos y envía.
9. La vista principal manda los datos al controlador para ser verificados.
10. El controlador verifica los datos y los envía a la base de datos para ser actualizados.
11. La base de datos actualiza la información de la inscripción y envía una confirmación al controlador.
12. El controlador lista las inscripciones vigentes en la vista principal para el administrador o integrante de grupo.
13. El administrador o integrante de grupo puede duplicar las inscripciones vigentes.
14. La vista principal le despliega el formulario con las opciones de la inscripción para su edición por parte del administrador o integrante de grupo que guarda los datos y envía.
15. La vista principal manda los datos al controlador para ser verificados.
16. El controlador verifica los datos y los envía a la base de datos para ser actualizados.
17. La base de datos almacena la información de la inscripción y envía una confirmación al controlador.
18. El controlador lista las inscripciones vigentes en la vista principal para el administrador o integrante de grupo.
19. El administrador o integrante de grupo puede eliminar una inscripción en el botón de eliminar una inscripción de la vista principal.
20. La vista principal manda los datos al controlador para ser verificados.
21. El controlador verifica los datos y los envía a la base de datos para ser actualizados.

22. La base de datos actualiza y envía una confirmación al controlador.
23. El controlador lista las inscripciones vigentes en la vista principal para el administrador o integrante de grupo.
24. El administrador o integrante de grupo puede agregar un requerimiento para la inscripción.
25. La vista principal le despliega el formulario con las opciones para agregar un tipo de requerimiento que necesite la inscripción por parte del administrador o integrante de grupo que guarda los datos y envía.
26. La vista principal manda los datos al controlador para ser verificados.
27. El controlador verifica los datos y los envía a la base de datos para ser almacenados.
28. La base de datos actualiza la información del requerimiento de la inscripción y envía una confirmación al controlador.
29. El controlador lista las inscripciones vigentes en la vista principal para el administrador o integrante de grupo.

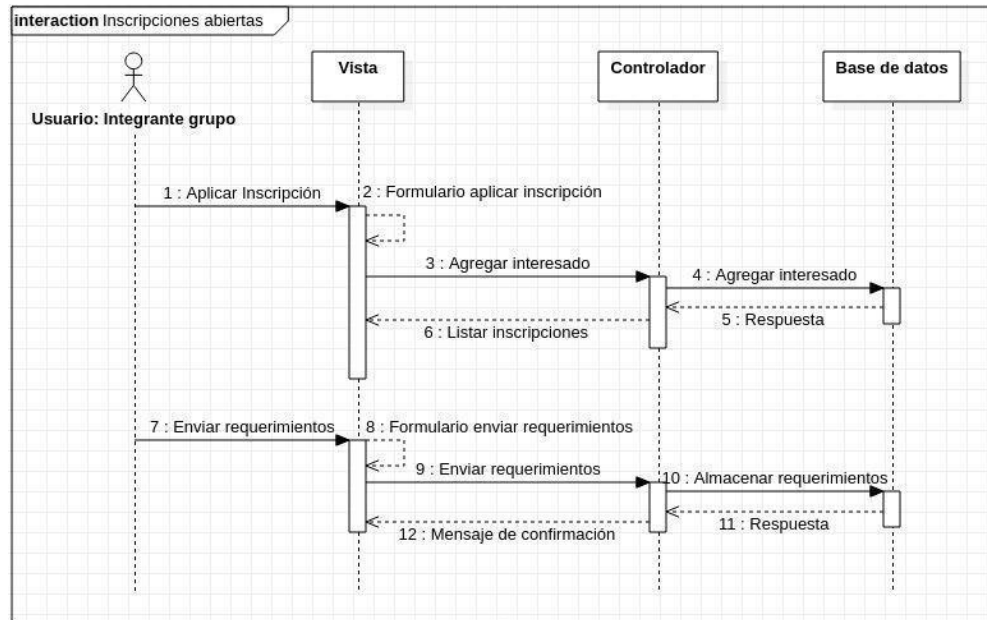
Figura 22. Diagrama de secuencia: Inscripciones abiertas por un usuario de la comunidad de la escuela y comunidad UIS



1. El usuario puede aplicar a la inscripción que sea de su interés en la vista principal.
2. La vista principal le despliega un formulario para que el usuario digite los datos y envíe la inscripción.
3. La vista principal envía los datos al controlador para que se validen.
4. El Controlador valida que los datos sean correctos y envía los datos a la base de datos
5. La base de datos guarda la información y da una respuesta de OK al Controlador.
6. El controlador lista las inscripciones en la Vista Principal.
7. El usuario puede consultar las inscripciones que sean de su interés en la vista principal.

8. La vista principal le despliega un formulario para que el usuario digite los datos y consulte la inscripción.
9. La vista principal envía los datos al controlador para que se validen.
10. El Controlador valida que los datos sean correctos y envía los datos a la base de datos
11. La base de datos consulta los datos y da una respuesta al Controlador.
12. El controlador lista las inscripciones en la Vista Principal.
13. El usuario envía los requerimientos de la inscripción en la vista principal.
14. La vista principal le despliega un formulario para que el usuario digite los datos y envíe los requerimientos de la inscripción.
15. La vista principal envía los datos al controlador para que se validen.
16. El Controlador valida que los datos sean correctos y envía los datos a la base de datos
17. La base de datos guarda la información y da una respuesta de OK al Controlador.
18. El controlador manda un mensaje de confirmación a la Vista Principal.

Figura 23. Diagrama de secuencia: Inscripciones abiertas para integrante del grupo.

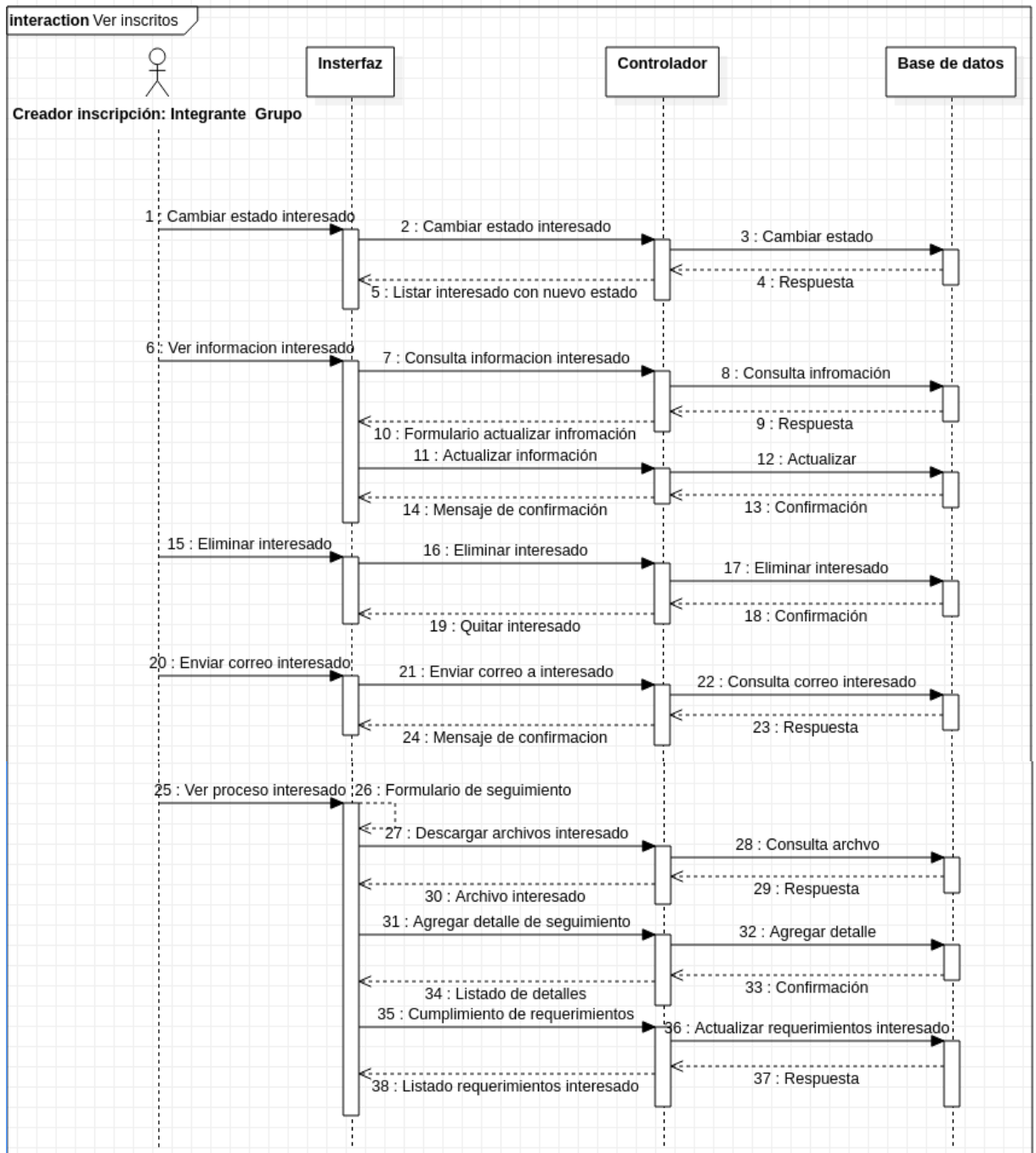


1. El integrante del grupo puede aplicar a una inscripción que desee en la vista principal.
2. La vista principal despliega un formulario que el integrante de grupo debe digitar y enviar en el botón de la vista principal.
3. La vista principal envía los datos al controlador.
4. El Controlador valida que los datos sean correctos y envía a la base de datos.
5. La base de datos almacena al inscrito y envía una respuesta al controlador.
6. El controlador recarga y lista las inscripciones vigentes en la vista principal.
7. El integrante del grupo puede enviar un requerimiento de una inscripción a la cual se encuentre registrado en la vista principal.
8. La vista principal despliega un formulario que el integrante de grupo debe digitar y enviar en el botón de la vista principal.
9. La vista principal envía los datos al controlador.
10. El Controlador valida que los datos sean correctos y envía a la base de datos.
11. La base de datos almacena los requerimientos del inscrito y envía una respuesta al controlador.

12. El controlador envía un mensaje de confirmación en la vista principal.

## Administrar inscritos o gestión de inscritos

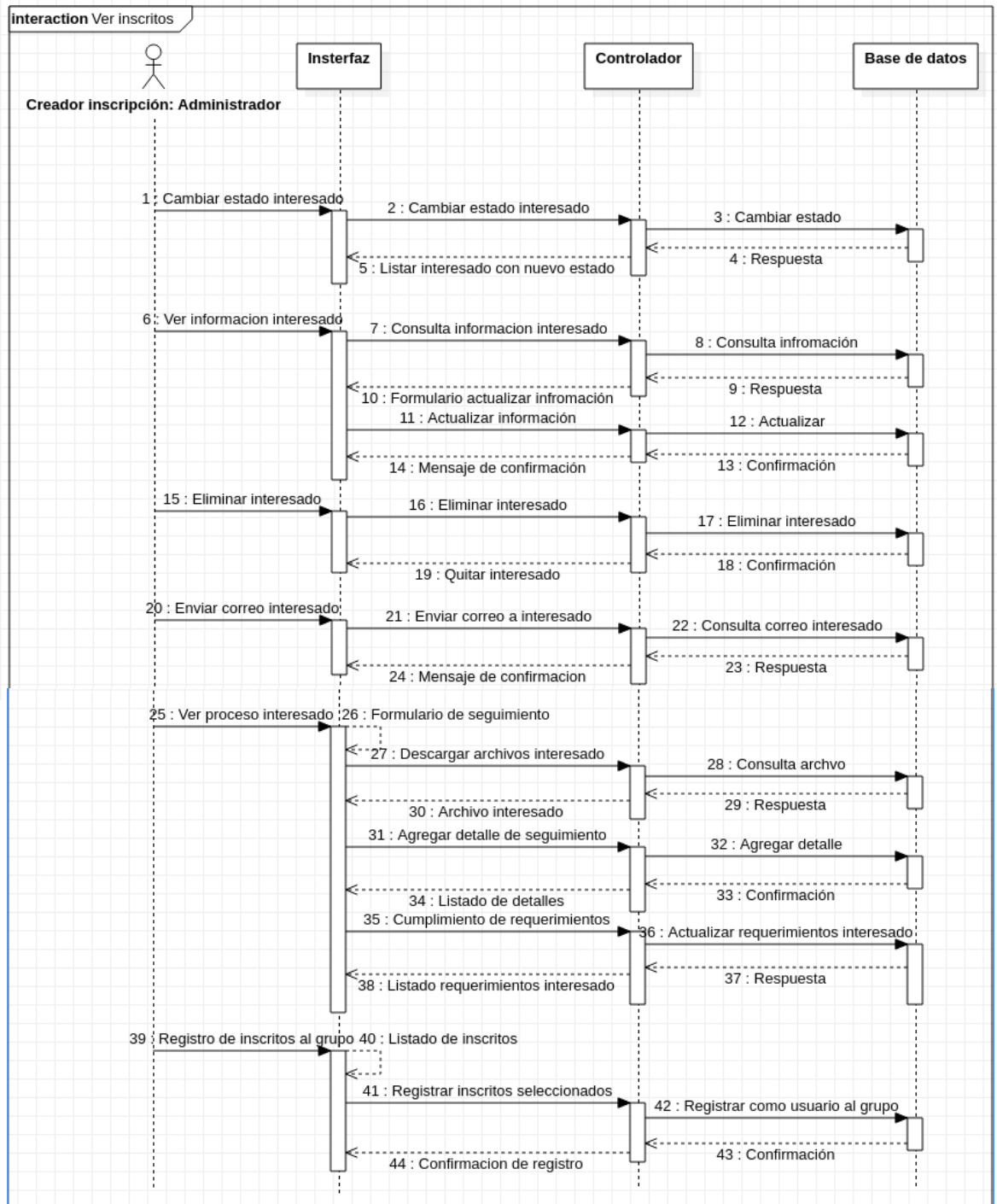
Figura 24. Diagrama de secuencia: Ver inscritos por integrante del grupo creador de la inscripción.



1. El creador de la inscripción cambia el estado del interesado el cual puede ser: Interesado, Preinscrito e Inscrito.
2. El controlador realiza la petición de actualización de estado a la base de datos.
3. En base de datos se actualiza el estado del interesado.
4. Retorna una respuesta de confirmación, al controlador.
5. El controlador retorna el nuevo estado del interesado a la interfaz el cual lista al interesado en la lista que corresponde al nuevo estado del mismo y eliminándolo del anterior estado.
6. El creador de la inscripción solicita ver la información del interesado.
7. El controlador realiza dicha solicitud a la base de datos.
8. En base de datos se realiza la consulta de la información.
9. Retorna una respuesta con la consulta realizada al controlador.
10. El controlador envía la información consultada a la interfaz la cual es representada mediante un formulario.
11. El creador de la inscripción puede actualizar esta información.
12. El controlador envía una petición de actualización a la base de datos, junto con la nueva información digitada.
13. En base de datos se actualiza la información y retorna una respuesta al controlador
14. El controlador envía una respuesta a la interfaz mediante un mensaje de confirmación el cual es mostrado al usuario.
15. El creador de la inscripción puede eliminar un interesado de la inscripción.
16. Se envía la información necesaria del interesado al controlado.
17. El controlador realiza la solicitud de eliminación del interesado.
18. En base de datos se elimina el interesado, y envía una respuesta al controlador.
19. El controlador envía la confirmación a la interfaz la cual borra al interesado de la lista.
20. El creador de la inscripción puede enviar un correo a uno o varios interesados.
21. Desde la interfaz se envía la información necesaria de los interesados al controlador.
22. El controlador realiza una petición para consultar el correo de el / los interesados.
23. En base de datos se realiza la consulta y envía una respuesta al controlador con dicha información.

24. El controlador recibe la información y ejecuta la acción de enviar el correo a los interesados, y envía un mensaje de confirmación a la interfaz.
25. El creador de la inscripción puede ver el proceso del interesado.
26. En la interfaz se muestra un formulario de seguimiento.
27. El interesado puede descargar archivos que han sido cargado por el interesado.
28. El controlador envía la información necesaria del archivo del interesado a base de datos para consultar la ruta de descarga.
29. En base de datos se realiza la consulta y envía una respuesta al controlador.
30. El controlador accede a la ruta del archivo a descargar y lo envía a la interfaz para que sea mostrado al creador de la inscripción.
31. El creador de la inscripción puede agregar un ítem de seguimiento al interesado.
32. El controlador realiza la petición para la inserción de dicho ítem a base de datos.
33. En base de datos se realiza la inserción y envía una respuesta al controlador
34. El controlador envía un mensaje de confirmación a la interfaz para que sea mostrado al usuario.
35. El creador de la inscripción puede verificar que los requerimientos de la inscripción se cumplan por parte del interesado.
36. El controlador realiza una petición para actualizar el listado de requerimientos de la inscripción por parte del interesado.
37. En base de datos se actualiza el listado de requerimientos y envía un mensaje de confirmación al controlador.
38. El controlador envía el mensaje de confirmación a la interfaz para que sea mostrado al usuario.

Figura 25. Ver inscritos escuela y ver inscritos UIS por el administrador e integrante del grupo.



1. Administrador puede cambiar el estado del interesado el cual puede ser: Interesado, Preinscrito e Inscrito.
2. El controlador realiza la petición de actualización de estado a la base de datos.
3. En base de datos se actualiza el estado del interesado.
4. Retorna una respuesta de confirmación, al controlador.
5. El controlador retorna el nuevo estado del interesado a la interfaz el cual lista al interesado en la lista que corresponde al nuevo estado del mismo y eliminándolo del anterior estado.
6. El administrador solicita ver la información del interesado.
7. El controlador realiza dicha solicitud a la base de datos.
8. En base de datos se realiza la consulta de la información.
9. Retorna una respuesta con la consulta realizada al controlador.
10. El controlador envía la información consultada a la interfaz la cual es representada mediante un formulario.
11. El administrador puede actualizar esta información.
12. El controlador envía una petición de actualización a la base de datos, junto con la nueva información digitada.
13. En base de datos se actualiza la información y retorna una respuesta al controlador
14. El controlador envía una respuesta a la interfaz mediante un mensaje de confirmación el cual es mostrado al usuario.
15. El administrador puede eliminar un interesado de la inscripción.
16. Se envía la información necesaria del interesado al controlado.
17. El controlador realiza la solicitud de eliminación del interesado.
18. En base de datos se elimina el interesado, y envía una respuesta al controlador.
19. El controlador envía la confirmación a la interfaz la cual borra al interesado de la lista.
20. El administrador puede enviar un correo a uno o varios interesados.
21. Desde la interfaz se envía la información necesaria de los interesados al controlador.
22. El controlador realiza una petición para consultar el correo de el / los interesados.
23. En base de datos se realiza la consulta y envía una respuesta al controlador con dicha información.
24. El controlador recibe la información y ejecuta la acción de enviar el correo a los interesados, y envía un mensaje de confirmación a la interfaz.

25. El creador de la inscripción puede ver el proceso del interesado.
26. En la interfaz se muestra un formulario de seguimiento.
27. El interesado puede descargar archivos que han sido cargado por el interesado.
28. El controlador envía la información necesaria del archivo del interesado a base de datos para consultar la ruta de descarga.
29. En base de datos se realiza la consulta y envía una respuesta al controlador.
30. El controlador accede a la ruta del archivo a descargar y lo envía a la interfaz para que sea mostrado al creador de la inscripción.
31. El creador de la inscripción puede agregar un ítem de seguimiento al interesado.
32. El controlador realiza la petición para la inserción de dicho ítem a la base de datos.
33. En base de datos se realiza la inserción y envía una respuesta al controlador
34. El controlador envía un mensaje de confirmación a la interfaz para que sea mostrado al usuario.
35. El creador de la inscripción puede verificar que los requerimientos de la inscripción se cumplan por parte del interesado.
36. El controlador realiza una petición para actualizar el listado de requerimientos de la inscripción por parte del interesado.
37. En base de datos se actualiza el listado de requerimientos y envía un mensaje de confirmación al controlador.
38. El controlador envía el mensaje de confirmación a la interfaz para que sea mostrado al usuario.
39. El administrador accede a la pestaña de registrar inscritos como usuarios del grupo.
40. En la interfaz se muestra el listado de interesados que se encuentran en estado INSCRITO.
41. El administrador selecciona uno o varios inscritos para ser registrados en el grupo con su respectivo tipo de usuario.
42. El controlador obtiene la información necesaria de los inscritos para ser registrados en el grupo y la envía a base de datos.
43. En base de datos se realiza el registro de el / los inscritos y retorna un mensaje de confirmación.
44. El controlador retorna el mensaje de confirmación a la interfaz para que sea mostrado al administrador.

